2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006

KTH Information and Communication Technology

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 1: Introduction

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapters 1-5

KTH Information and Communication Technology

Welcome to the Internetworking course!

The course should be fun.

We will dig deeper into the TCP/IP protocols and protocols built upon them.

Information about the course is available from the course web page:

`http://www.it.kth.se/courses/2G1305/`

Note that the above URL will change - due to the reoganization of KTH to:

`http://www.cos.ict.kth.se/education/msc/ccs/courses/2G1305/`

Maguire
maguire@it.kth.se

Welcome to the Internetworking course!
2006.03.13

Introduction 3 of 74
Internetworking/Internetteknik

Staff Associated with the Course

Instructor (Kursansvarig)

prof. Gerald Q. Maguire Jr. <maguire@it.kth.se>

Assistants for Recitation Sessions (Övningar)

To be annouced

Administrative Assistant: recording of grades, registration, etc.

Irina Radulescu <irina.radulescu at wireless.kth.se>

Maguire
maguire@it.kth.se

Staff Associated with the Course
2006.03.13

Introduction 4 of 74
Internetworking/Internetteknik

Goals, Scope and Method

Goals of the Course

- To give deep knowledge and competence (*designing*, *analyzing,* and *developing*) of Internet protocols and architecture, both practical and analytical.

- To be able to read and understand the Internet standardization documents (IETF RFCs and Internet Drafts) and current Internet literature.

- You should have the knowledge and competence to do exciting Internet related research and development.

Scope and Method

- Dig deeper into the TCP/IP protocol suite by using diagnostic tools to examine, observe, and analyze these protocols in action. Understanding the details!

- Demonstrate this by writing a written report.

Maguire
maguire@it.kth.se

Goals, Scope and Method
2006.03.13

Introduction 5 of 74
Internetworking/Internetteknik

Aim

After this course you should be able to read the current internet literature at the level of IEEE Communications Magazine, IEEE Network, IEEE Transactions on Communications, IEEE Transactions on Communications, IEEE Journal on Selected Areas in Communications, IEEE/ACM Transactions on Networking, IEEE Communications Surveys (On-line Journal), … . See the IEEE Communication Society's *list of publications*.

While you may not be able to understand all of the articles in the above journals and magazines, you should be able to read 90% or more of the articles and have good comprehension. *You should develop a habit of reading the journals, trade papers, etc.*

You should be able to write internetworking articles at the level of Miller Freeman's *Network Magazine* or IEEE Internet Computing. In subsequent courses you will also develop you ability to orally present your ideas.

Maguire
maguire@it.kth.se

Aim
2006.03.13

Introduction 6 of 74
Internetworking/Internetteknik

Prerequisites

- Datorkommunikation och datornät/Data and computer communication
  or

- Equivalent knowledge in Computer Communications (this requires permission of the instructor)

Maguire
maguire@it.kth.se

Prerequisites
2006.03.13

Introduction 7 of 74
Internetworking/Internetteknik

Contents

This course will focus on the **protocols** that are the fundaments of the Internet. We will explore what internetworking means and what it requires. We will give both practical and more general knowledge concerning the Internet network architecture.

The course consists of 14 hours of lectures and 10 hours of recitations (övningar) [possible some laboratory exercises].

Maguire
maguire@it.kth.se

Contents
2006.03.13

Introduction 8 of 74
Internetworking/Internetteknik

Topics

- What an internet is and what is required of protocols to allow internetworking
- details of routing and routing protocols (RIP, BGP, OSPF, …)
- multicasting
- Domain Name System (DNS, Dynamic DNS)
- what happens from the time a machine boots until the applications are running (RARP, BOOTP, DHCP, TFTP)
- details of the TCP protocols and some performance issues
- details of a number of application protocols (especially with respect to distributed file systems)
- network security (including firewalls, AAA, IPSec, SOCKs, …)
- differences between IPv6 and IPv4
- network management (SNMP) and
- We will also examine some emerging topics:
  - cut-through routing, tag switching, flow switching, QoS, Mobile IP, Voice over IP, SIP, NAT, VPN, Diffserv, … .

Maguire
maguire@it.kth.se

Topics
2006.03.13

Introduction 9 of 74
Internetworking/Internetteknik

Examination requirements

- Written assignment (4 p)
  - based on lectures, recitations, and your references

Grades: U, 3, 4, 5

Maguire
maguire@it.kth.se

Examination requirements
2006.03.13

Introduction 10 of 74
Internetworking/Internetteknik

Written Assignment

Goal: to gain analytical or practical experience and to show that you have mastered some Internetworking knowledge (in addition to what you show on the written examination).

- Can be done in a group of **1 to 3** students (formed by yourself). Each student must contribute to the final report.
- There will be one or more suggested topics, additional topics are possible (discuss this with one of the teachers **before** starting).

Maguire
maguire@it.kth.se

Written Assignment
2006.03.13

Introduction 11 of 74
Internetworking/Internetteknik

Assignment Registration and Report

- Registration: March 31, 2006, to <maguire@it.kth.se>
  - Group members, leader.
  - Topic selected.

- For Analytical Assignments
  - The length of the final report should be 7-8 pages (roughly 3,000 words) for each student.
  - Contribution by each member of the group - must be clear

- For Practical Assignments
  - A short technical document describing: 1) what you have done; 2) who did what; 3) methods and tools used; 3) the test or implementation results.

Final Report: **May 12, 2006**

- Send email with URL link to a PDF file to <maguire@it.kth.se>

- Late assignments will <u>not</u> be accepted (i.e., there is no guarantee that they will graded in time for the end of the term)

Note that it is pemissible to start working *well in advance* of the deadlines!

Maguire
maguire@it.kth.se

Assignment Registration and Report
2006.03.13

Introduction 12 of 74
Internetworking/Internetteknik

Literature

The course will mainly be based on the book: Behrouz A. Forouzan, *TCP/IP Protocol Suite*, 3rd edition, McGraw-Hill, publication date January 2005, (Copyright 2006) 896 pages, ISBN 0072967722 (hardbound) or 0071115838 (softbound)

Other additional references include:

- *TCP/IP Illustrated,* `Volume 1: The Protocols` by `W. Richard Stevens`, Addison-Wesley, 1994, ISBN 0-201-63346-9 and *Internetworking with TCP/IP: Principles, Protocols, and Architectures, Vol. 1,* by Douglas E. Comer, Prentice Hall, 4th edt. 2000, ISBN 0-13-018380-6.

- the commented source code in *TCP/IP Illustrated,* `Volume 2: The Implementation` by Gary R. Wright and W. Richard Stevens, Addison-Wesley, 1995, ISBN 0-201-63354-X

- *IPv6: The New Internet Protocol*, by Christian Huitema, Prentice-Hall, 1996, ISBN 0-13-241936-X.

Maguire
maguire@it.kth.se

Literature
2006.03.13

Introduction 13 of 74
Internetworking/Internetteknik

- concerning HTTP we will refer to *TCP/IP Illustrated,* `Volume 3: TCP for Transactions, HTTP, NNTP, and the UNIX Domain Protocols`, Addison-Wesley, 1996, ISBN 0-201-63495-3.

With regard to Mobile IP the following two books are useful as additional sources:

- *Mobile IP: Design Principles and Practices* by Charles E. Perkins, Addison-Wesley, 1998, ISDN 0-201-63469-4.

- *Mobile IP: the Internet Unplugged* by James D. Solomon, Prentice Hall, 1998, ISBN 0-13-856246-6.

Internetworking Technologies Handbook by Kevin Downes (Editor), H. Kim Lew, Steve Spanier, Tim Stevenson (Online:

`http://www-fr.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/index.htm`)

We will refer to other books, articles, and RFCs as necessary. In addition, there will be **compulsory** written exercises.

Maguire
maguire@it.kth.se

Literature
2006.03.13

Introduction 14 of 74
Internetworking/Internetteknik

Lecture Plan

Subject to revision!

- Lecture 1:   Introduction and IP basics
- Lecture 2:   IP and ICMP
- Lecture 3:   User Datagram Protocol (UDP) & TCP
- Lecture 4:   TCP and SCTP
- Lecture 5:   Dynamic Routing
- Lecture 6:   IP Multicast and Mobile IP
- Lecture 7:   Internet Security, VPNs, Firewalls, and NAT
              Future Issues and Summary

Maguire
maguire@it.kth.se

Lecture Plan
2006.03.13

Introduction 15 of 74
Internetworking/Internetteknik

Context of the course

"The network called the Internet is the single most important development in the communications industry since the public switched voice network was constructed…"

-- John Sidgmore
      CEO, UUNET Technologies

      and COO, WorldCom[1]

---

1. http://www.lucent.com/enterprise/sig/exchange/present/slide2.html {this URL no longer functions}

Maguire
maguire@it.kth.se

Context of the course
2006.03.13

Introduction 16 of 74
Internetworking/Internetteknik
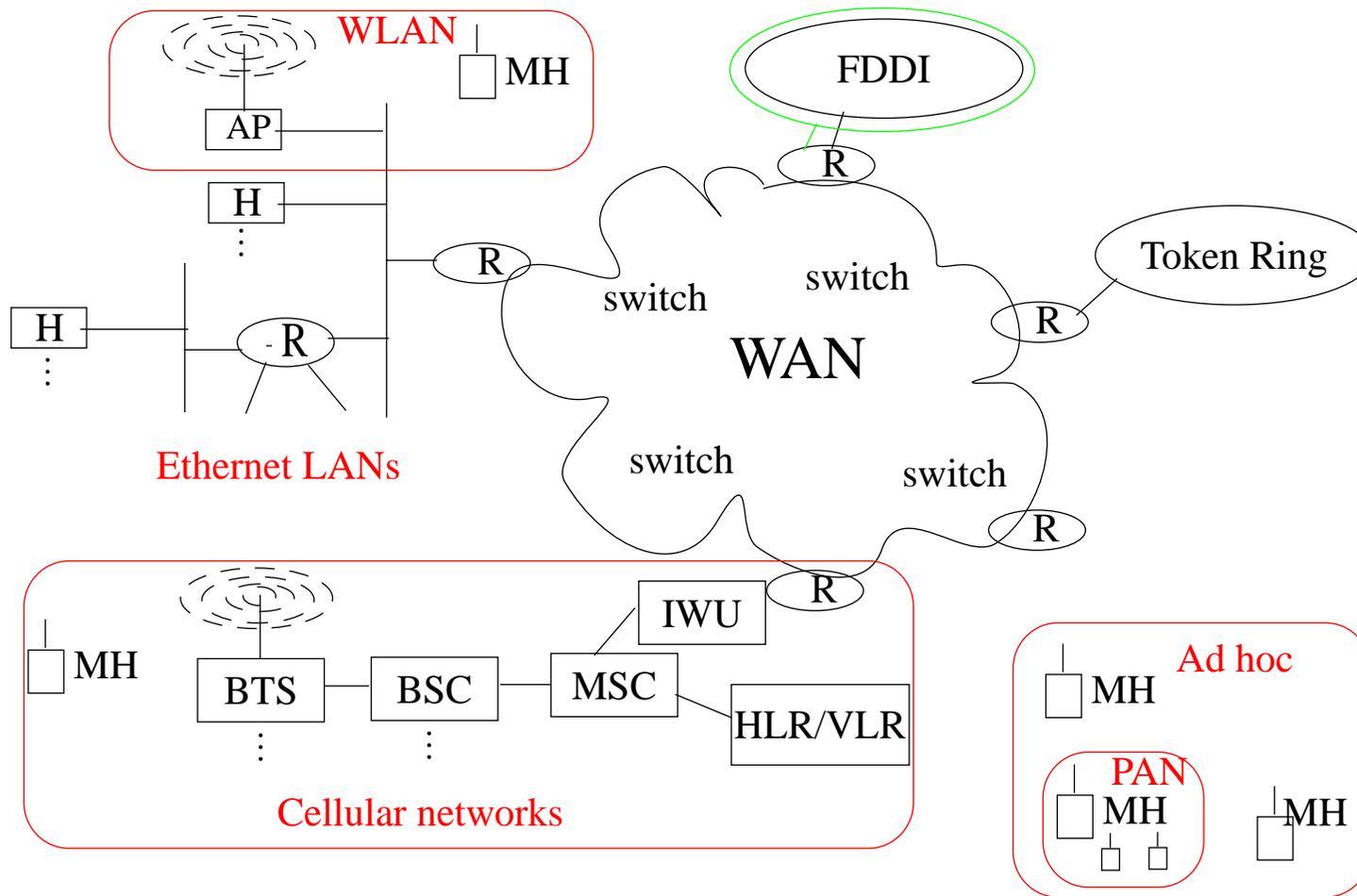
Network Architecture

Figure 1: Multiple network technologies - internetworked together

Note that some of the routers act as **gateways** between different types of networks.

Maguire
maguire@it.kth.se

Network Architecture
2006.03.13

Introduction 17 of 74
Internetworking/Internetteknik

Power of the Internet (chaos)

"'Historically, the Internet has been an environment in which to experiment. There have been a few basic rules. The most important is the standard for IP and TCP.

The are other important standards for promulgating routing information and the like, but the real power of the Internet idea is that there are not mandidated stanards for what can run over the 'Net.

Anyone who adheres to TCP/IP standards can create applications and run them without getting anyone's permission. No ISP even has to know you are experimenting (or playing, which is also OK). This freedom produces unpredictable results. New industries can be created almost overnight and existing industries severely affected. …"

-- Scott O. Bradner, "The Importance of Being a Dynamist", Network World, December 13, 1999, p. 48 (www.nwfusion.com)

Maguire
maguire@it.kth.se

Power of the Internet (chaos)
2006.03.13

Introduction 18 of 74
Internetworking/Internetteknik

Internet Trends

- Numbers of users and internet devices increases very rapidly
  - Network Wizards' Internet Domain Survey - *http://www.isc.org/index.pl?/ops/ds/*
    - Jan.2006   394,991,609 hosts
    - Jan. 2005: 317,646,084 hosts
  - RIPE's survey *European hosts* :
    - Jan. 2005: 25,389,171 real hosts of which 1,521,328 are in Sweden vs.
    - {Note that the above estimates are based on DNS information}
  - Network Weather Maps - *http://www.cybergeography.org/atlas/weather.html*
    http://www.nordu.net/stat-q/load-map/ndn-map,,traffic,busy

- QoS: Demand for integrating many different types of traffic, such as video, audio, and data traffic, into one network $\Rightarrow$ **Multicast**, **IPv6**, **RSVP**, **DiffServ**, emphasis on **high performance**, and TCP **extensions (**we will examine a number of these in this course)

- Mobility: both users and devices are mobile
  - There is a difference between portable (bärbar) vs. mobile (mobil).
  - IP is used in wireless systems (for example 3G cellular).
  - Increasing use of wireless in the last hop (WLAN, PAN, Wireless MAN, …)

- Security:
  - Wireless mobile Internet - initial concern driven by wireless link
  - Fixed Internet - distributed denial of service attacks, increasing telecommuting, …

Maguire
maguire@it.kth.se

Internet Trends
2006.03.13

Introduction 19 of 74
Internetworking/Internetteknik

IP traffic growing exponentially!

Traffic increasing (but not due to voice)

- IP traffic between US and Sweden many times the total voice+FAX traffic

- many Gbit/s transatlantic fiber

Fixed Links - arbitrarily fast:

- LANs: 10Mbits/s, 100Mbits/s, 1Gbits/s, 10Gbits/s, …

- Backbones: Gigabits/s
  Transoceanic fibers between continents $\Rightarrow$ Gbit/s $\Rightarrow$ Tbit/s

- Major sites link to backbones: increasingly 10^+Mbit/s to Gbit/s

- Individual users links: 28.8 Kbits/s and ISDN (128Kbits/s)
  $\Rightarrow$ ethernet and xDSL (2 Mbits/s .. 100 Mbits/s)

Points of Presence (PoPs) + FIX/CIX/GIX/MAE[1] $\Rightarrow$ GigaPoPs

(George) Guilder's Law states that network speeds will triple every year for the next 25 years.
This dwarfs Moore's law that predicts CPU processor speed will double every 18 months.

---

1. Federal Internet eXchange (FIX), Commercial Internet eXchange (CIX), Global Internet eXchange (GIX),   Metropolitan Access Exchange (MAE)

Maguire
maguire@it.kth.se

IP traffic growing exponentially!
2006.03.13

Introduction 20 of 74
Internetworking/Internetteknik

Speed

"... The Internet world moves fast. The integration of voice and data onto a single network is not being lead by the International Telecommunications Union or by Bellcore. Rather, its being lead by entrepreneurs like …. Until now, the voice networks dominated. Data could ride on top of the phone network -- when it was convenient. The explosion of data networking and Internet telephony technology is making the opposite true. Now voice can ride on data networks -- when it is convenient."[1]

Because of bandwidth constraints, Internet telephony would not be a major factor **"for a long time -- maybe nine to twelve months."**

-- president of a major ISP[2]

Internet time - 7x real time

-- Ira Goldstein, HP

_____

1. from http://www.dialogic.com/solution/internet/apps.htm {no longer a valid URL}

2. from http://www.dialogic.com/solution/internet/apps.htm {no longer a valid URL}

Maguire
maguire@it.kth.se

Speed
2006.03.13

Introduction 21 of 74
Internetworking/Internetteknik

Growth rates

Some people think the Internet bandwidth explosion is relatively recent, but right from the beginning it's been a race against an ever-expanding load. It isn't something you can plan for. In fact, the notion of long-range planning like the telcos do is almost comical. Just last month, a local carrier asked us why we didn't do five-year plans, and we said, "We do-about once a month!"

-- Mike O'Dell[1] VP and Chief Technologist UUNET

Mike points out that the growth rate of the Internet is driven by the increasing speed of computers, while telcos have traffic which was proportional to the growth in numbers of people (each of whom could only use a very small amount of bandwidth).

- by 1997 UUNET was adding at least one T3/day to their backbone

---

1. from http://www.data.com/25years/mike_odell.html {no longer a valid URL}

Maguire
maguire@it.kth.se

Growth rates
2006.03.13

Introduction 22 of 74
Internetworking/Internetteknik

¿Question?

"Which would you rather have twice as fast:
  your computer's processor or modem?

After 30 years of semiconductor doublings under Moore's Law, processor speed are measured in megahertz. On the other hand, after 60 years of telco's snoozing under monopoly law, modem speeds are measure in kilobits. Modems are way too slow for Internet access, but you knew that."[1]

-- Bob Metcalfe, inventor of Ethernet in 1973

1. "From the Ether: Moving intelligence and Java Packets into the Net will conserve bandwidth", by Bob Metcalfe, Inforworld, Oct., 6, 1997, pg. 171.

Maguire
maguire@it.kth.se

¿Question?
2006.03.13

Introduction 23 of 74
Internetworking/Internetteknik

Increasing Data Rates

"Ethernet'

- 3 Mbps Ethernet (actually 2.944 Mbits/sec)
- 10 Mbps Ethernet (which became IEEE 802.3)
- 100 Mbps Ethernet (100Tx)
- Gigabit Ethernet (IEEE 802.3z, IEEE 802.3ab)
- 10 Gbps Ethernet (IEEE 802.3ae)

Optical

- Dense Wavelength Division Multiplexing (DWDM) - allowing 1000s of multi-Gbits/s channels to be carried on existing fibers

Wireless

- IEEE 802.11 Wireless LAN (2 .. 54 Mbits/s)
- IEEE 802.15 Wireless Personal Area Network (WPAN)
- IEEE 802.16 Metropolitan Area Networks - Fixed Broadband Wireless (2-11; 10 .. 66 GHz)

Maguire
maguire@it.kth.se

Increasing Data Rates
2006.03.13

Introduction 24 of 74
Internetworking/Internetteknik

Internetworking

Internetworking is

- based on the interconnection (concatenation) of multiple networks
- accommodates multiple underlying hardware technologies by providing a way to interconnect **heterogeneous** networks and makes them inter-operate.
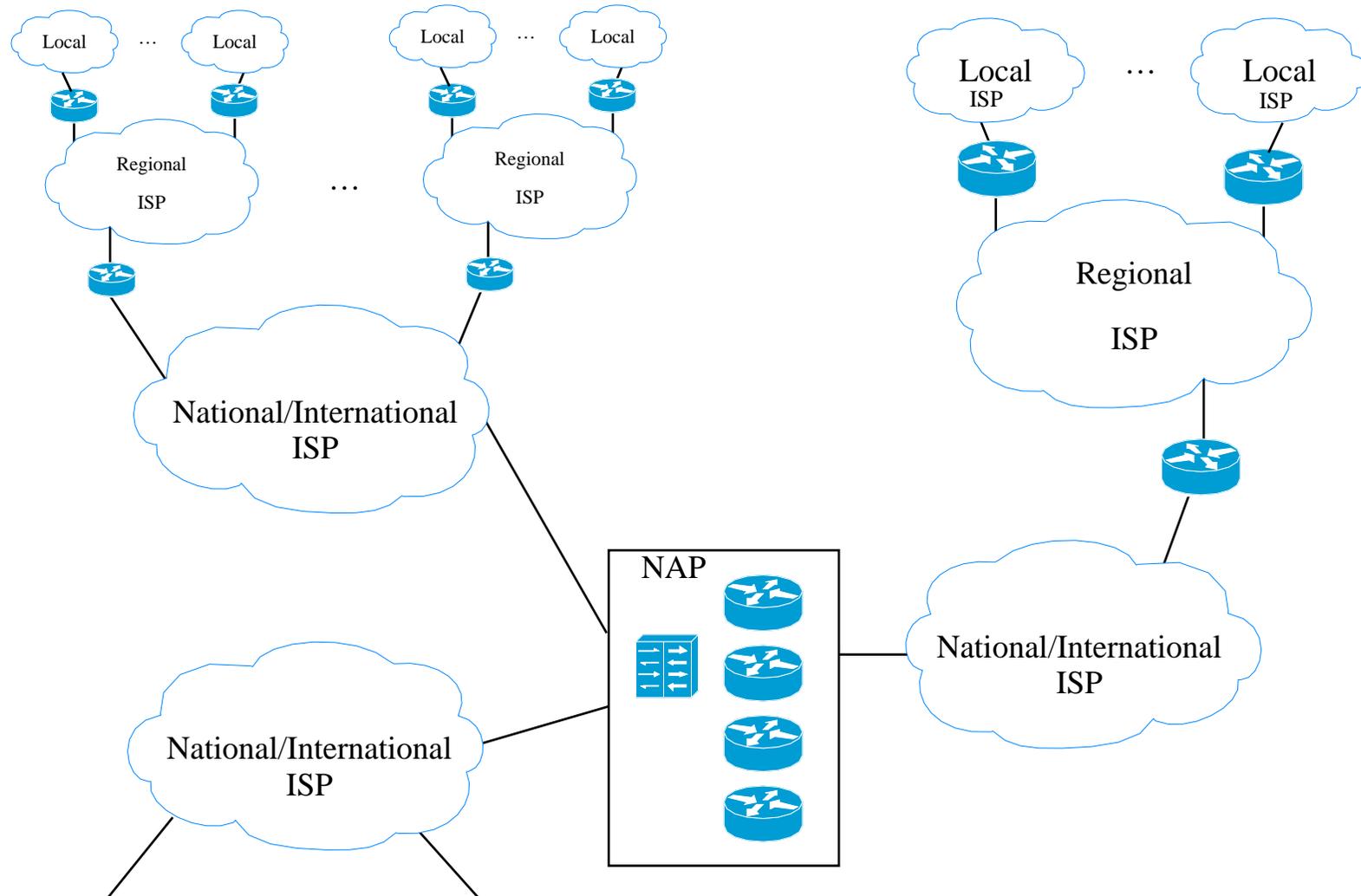
We will concern ourselves with one of the most common internetworking protocols IP (there *are* other internetworking protocols, such as Novell's Internetwork Packet Exchange (IPX), Xerox Network Systems (XNS), IBM's Systems Network Architecture (SNA), OSI's ISO-IP).

We will examine both IP:

- version 4 - which is in wide use
- version 6 - which is coming into use

Internet: the worldwide internet

Maguire
maguire@it.kth.se

Internetworking
2006.03.13

Introduction 25 of 74
Internetworking/Internetteknik

The Internet Today

key: Internet Serice Provider (ISP), Network Access Point (NAP)

Maguire
maguire@it.kth.se

The Internet Today
2006.03.13

Introduction 26 of 74
Internetworking/Internetteknik

Basic concepts

open-architecture
networking [1],[2]

- Each distinct network stands on its own makes its own technology choices, etc. ⇒ no changes within each of these networks in order to internet
- Based on best-effort delivery of datagrams
- Gateways interconnect the networks
- No global control

The End2End
Argument [4]

Some basic design principle for the Internet:
- Specific application-level functions should **not** be built into the lower levels
- Functions implemented **in** the network should be simple and general.
- Most functions are implemented (as software) at the edge
  ⇒ complexity of the core network is reduced
  ⇒ increases the chances that new applications can be easily added.

See also [5], [6]

Hourglass
(Stuttgart
wineglass) Model

- Anything over IP
- IP over anything

Note the broad (and open) top - enabling lots
and lots of application

Maguire
maguire@it.kth.se

Basic concepts
2006.03.13

Introduction 27 of 74
Internetworking/Internetteknik
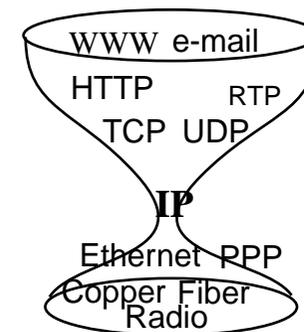
Review of Layering
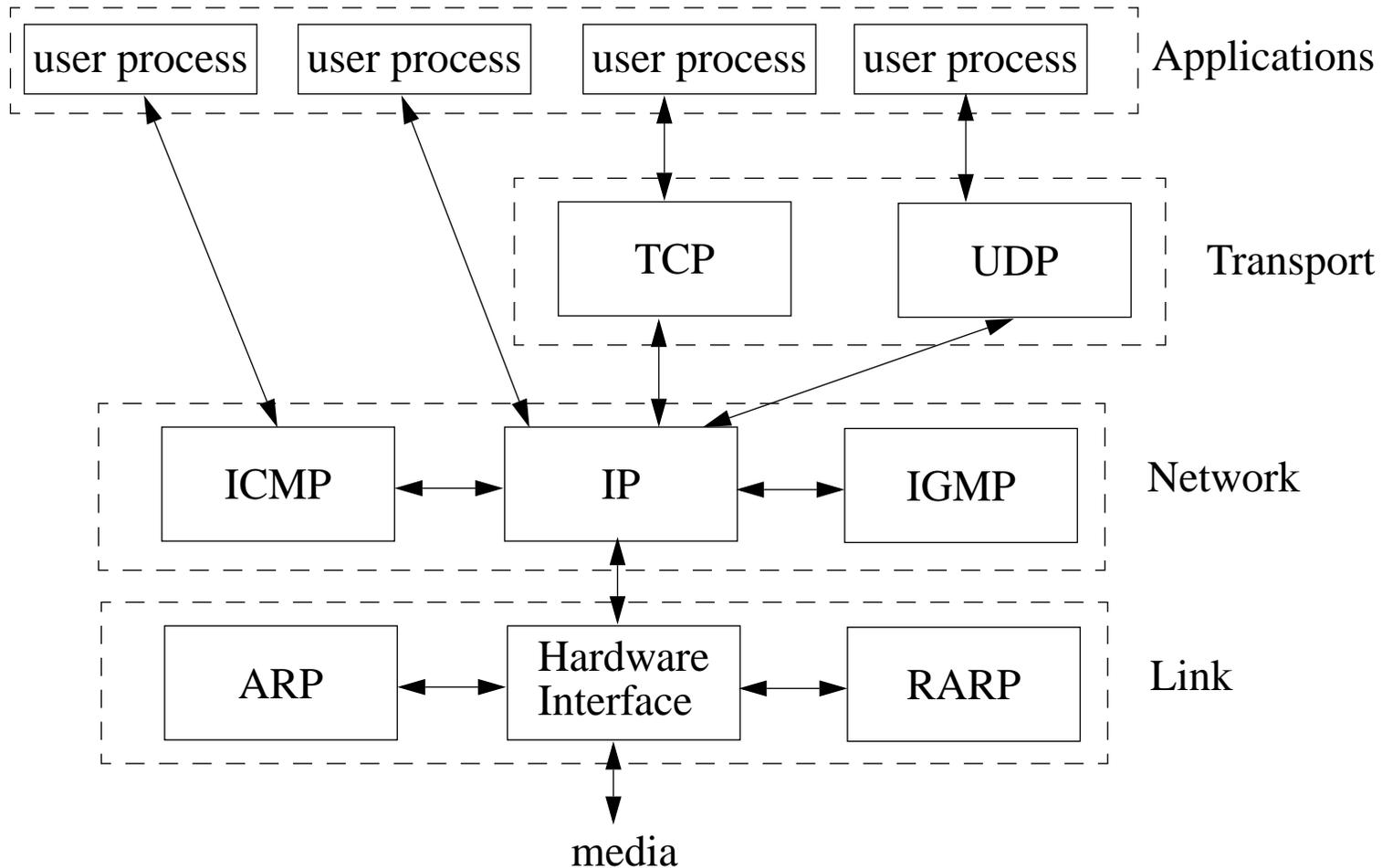
Figure 2: protocol layers in the TCP/IP protocol suite
(see Stevens, Volume 1, figure 1.4, pg. 6)

Maguire
maguire@it.kth.se

Review of Layering
2006.03.13

Introduction 28 of 74
Internetworking/Internetteknik

Encapsulation

	user data		Applications

Appl header	user data		

TCP header	application data		Transport

← 20 →

IP header	transport data		Network

← 20 →

Ethernet header	network data	Ethernet trailer	Link

← 14 → ← 46 to 1500 bytes → ← 4 →

Ethernet Frame
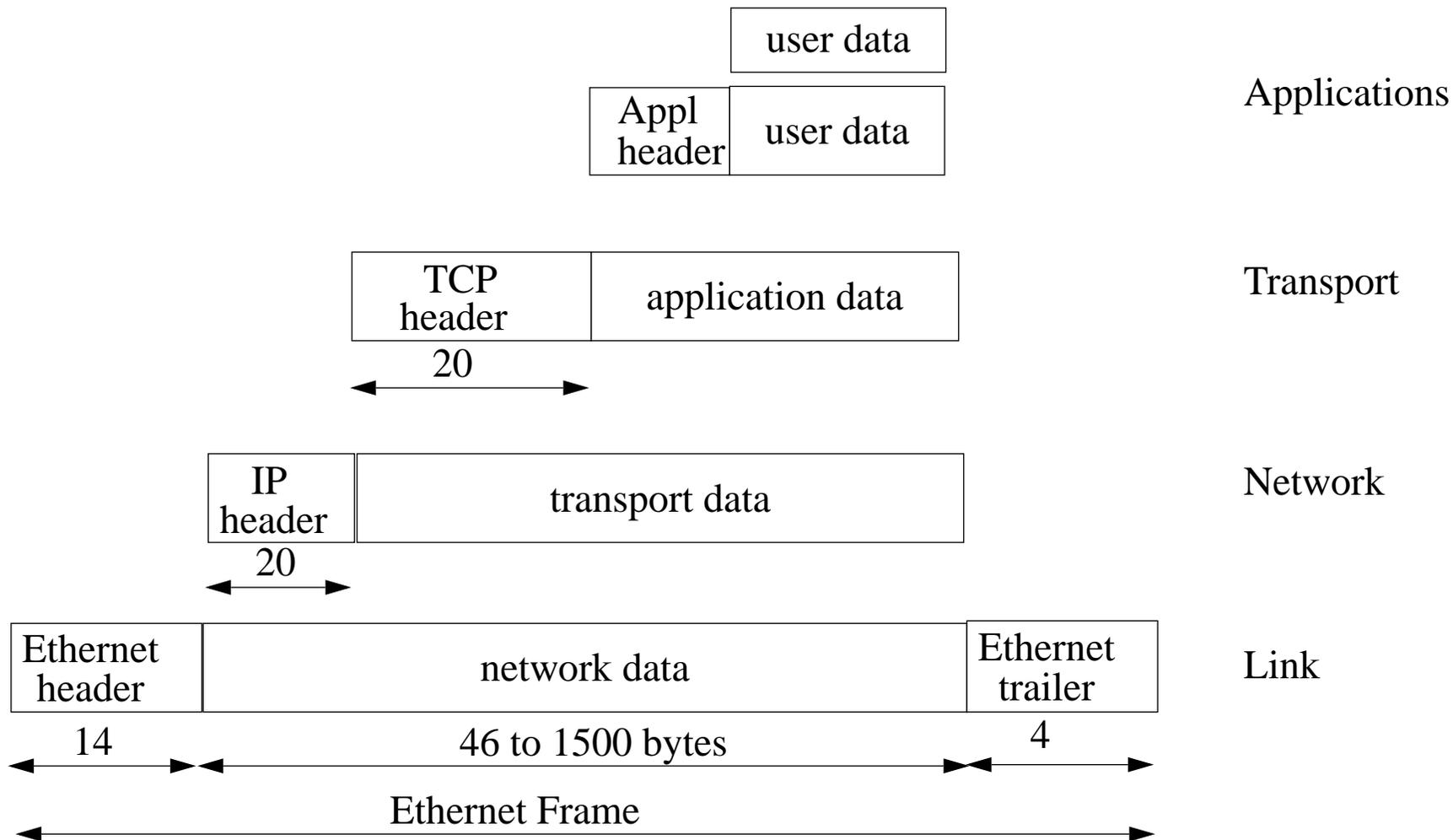
Figure 3: Encapsulation of data
(see Stevens, Volume 1, figure 1.7, pg. 10)

Maguire
maguire@it.kth.se

Encapsulation
2006.03.13

Introduction 29 of 74
Internetworking/Internetteknik

Demultiplexing

user process   user process   …   user process   user process …   user process …   user process …

TCP   UDP   TCP   UDP

Demux on
TCP or UDP
port number

ICMP   IGMP

IPv4

Demux on
protocol value
in IP header

IPv6

ARP   Driver   RARP

Demux on
Frame type and/or IP version

incoming frame - accepted by matching
address or multicast address

Figure 4: Demultiplexing
(adapted from Stevens, Volume 1, figure 1.8, pg. 11; with dual IP stacks)

Maguire
maguire@it.kth.se

Demultiplexing
2006.03.13

Introduction 30 of 74
Internetworking/Internetteknik

Addresses in TCP/IP

- ## Transport layer
  - Port number

- ## Network layer
  - IP address
  - Protocol

- ## Link & Physical layers
  - Frame type
  - Media Access and Control (MAC) address

Maguire
maguire@it.kth.se

Addresses in TCP/IP
2006.03.13

Introduction 31 of 74
Internetworking/Internetteknik

Internet Protocol version 4 (IPv4) (RFC 791)

0	7	8	15	16	23 24	31

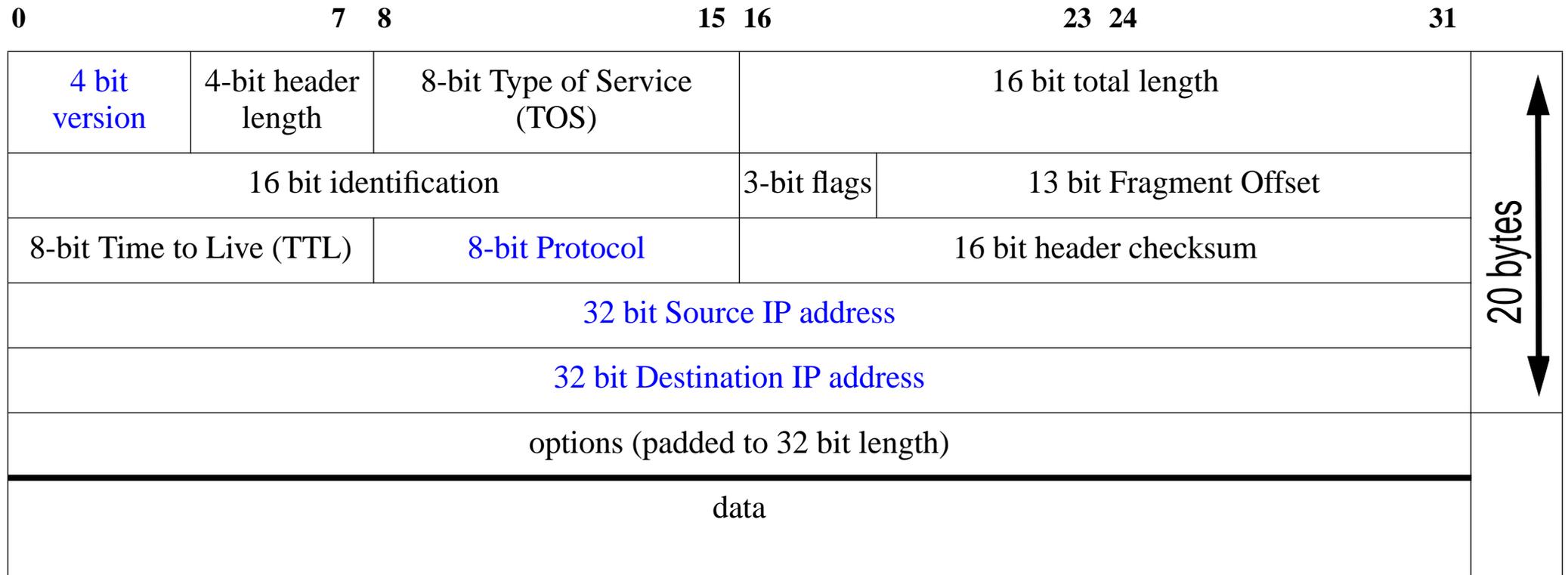| 4 bit version	4-bit header length	8-bit Type of Service (TOS)	16 bit total length			
16 bit identification			3-bit flags	13 bit Fragment Offset		
8-bit Time to Live (TTL)		8-bit Protocol	16 bit header checksum			
32 bit Source IP address						
32 bit Destination IP address						
options (padded to 32 bit length)						
data						

20 bytes

Figure 5: IP header (see Stevens, Vol. 1, figure 3.1, pg. 34)

The fields: Version, Protocol, and Source & Destination IP addresses are all used for demultiplexing the incoming IP packet.

We will first examine version 4, then later in the course version 6.

Maguire
maguire@it.kth.se

Internet Protocol version 4 (IPv4) (RFC 791)
2006.03.13

Introduction 32 of 74
Internetworking/Internetteknik

IP "Protocol" field (RFC 1700)

In the Internet Protocol (IP) [DDN], [RFC791] there is a field, called Protocol, to identify the next level protocol. This is an 8 bit field.

Assigned Internet Protocol Numbers (assigned by *Internet Assigned Numbers Authority* (IANA) *http://www.iana.org/assignments/protocol-numbers*

Decimal	Keyword	Protocol	References
0	HOPOPT	IPv6 Hop-by-Hop Option	[RFC1883]
1	ICMP	Internet Control Message	[RFC792]
2	IGMP	Internet Group Management	[RFC1112]
3	GGP	Gateway-to-Gateway	[RFC823]
4	IP	IP in IP (encapsulation)	[RFC2003]
5	ST	Stream	[RFC1190,RFC1819]
6	TCP	Transmission Control	[RFC793]
7	CBT	CBT	[Ballardie]
8	EGP	Exterior Gateway Protocol	[RFC888,DLM1]
9	IGP	any private interior (e.g., used by Cisco for their IGRP)	[IANA]
10	BBN-RCC-MON	BBN RCC Monitoring	[SGC]
11	NVP-II	Network Voice Protocol	[RFC741,SC3]
12	PUP	PUP	[PUP,XEROX]

Maguire
maguire@it.kth.se

IP "Protocol" field (RFC 1700)
2006.03.13

Introduction 33 of 74
Internetworking/Internetteknik

Decimal	Keyword	Protocol	References
13	ARGUS	ARGUS	[RWS4]
14	EMCON	EMCON	[BN7]
15	XNET	Cross Net Debugger	[IEN158,JFH2]
16	CHAOS	Chaos	[NC3]
17	UDP	User Datagram	[RFC768,JBP]
18	MUX	Multiplexing	[IEN90,JBP]
19	DCN-MEAS	DCN Measurement Subsystems	[DLM1]
20	HMP	Host Monitoring	[RFC869,RH6]
21	PRM	Packet Radio Measurement	[ZSU]
22	XNS-IDP	XEROX NS IDP	[ETHERNET,XEROX]
23	TRUNK-1	Trunk-1	[BWB6]
24	TRUNK-2	Trunk-2	[BWB6]
25	LEAF-1	Leaf-1	[BWB6]
26	LEAF-2	Leaf-2	[BWB6]
27	RDP	Reliable Data Protocol	[RFC908,RH6]
28	IRTP	Internet Reliable Transaction	[RFC938,TXM]
29	ISO-TP4	ISO Transport Protocol Class 4	[RFC905,RC77]
30	NETBLT	Bulk Data Transfer Protocol	[RFC969,DDC1]
31	MFE-NSP	MFE Network Services Protocol	[MFENET,BCH2]
32	MERIT-INP	MERIT Internodal Protocol	[HWB]
33	SEP	Sequential Exchange Protocol	[JC120]
34	3PC	Third Party Connect Protocol	[SAF3]
35	IDPR	Inter-Domain Policy Routing Protocol	[MXS1]

Maguire
maguire@it.kth.se

IP "Protocol" field (RFC 1700)
2006.03.13

Introduction 34 of 74
Internetworking/Internetteknik

Decimal	Keyword	Protocol	References
36	XTP	XTP	[GXC]
37	DDP	Datagram Delivery Protocol	[WXC]
38	IDPR-CMTP	IDPR Control Message Transport Proto	[MXS1]
39	TP++	TP++ Transport Protocol	[DXF]
40	IL	IL Transport Protocol	[Presotto]
41	IPv6	Ipv6	[Deering]
42	SDRP	Source Demand Routing Protocol	[DXE1]
43	IPv6-Route	Routing Header for IPv6	[Deering]
44	IPv6-Frag	Fragment Header for IPv6	[Deering]
45	IDRP	Inter-Domain Routing Protocol	[Sue Hares]
46	RSVP	Reservation Protocol	[Bob Braden]
47	GRE	General Routing Encapsulation	[Tony Li]
48	MHRP	Mobile Host Routing Protoco	[David Johnson]
49	BNA	BNA	[Gary Salamon]
50	ESP	Encap Security Payload for IPv6	[RFC1827]
51	AH	Authentication Header for IPv6	[RFC1826]
52	I-NLSP	Integrated Net Layer Security TUBA	[GLENN]
53	SWIPE	IP with Encryption	[JI6]
54	NARP	NBMA Address Resolution Protocol	[RFC1735]
55	MOBILE	IP Mobility	[Perkins]
56	TLSP	Transport Layer SecurityProtocol (using Kryptonet key management)	[Oberg]
57	SKIP	SKIP	[Markson]

Maguire
maguire@it.kth.se

IP "Protocol" field (RFC 1700)
2006.03.13

Introduction 35 of 74
Internetworking/Internetteknik

Decimal	Keyword	Protocol	References
58	IPv6-ICMP	ICMP for IPv6	[RFC1883]
59	IPv6-NoNxt	No Next Header for IPv6	[RFC1883]
60	IPv6-Opts	Destination Options for IPv6	[RFC1883]
61		any host internal protocol	[IANA]
62	CFTP	CFTP	[CFTP,HCF2]
63		any local network	[IANA]
64	SAT-EXPAK	SATNET and Backroom EXPAK	[SHB]
65	KRYPTOLAN	Kryptolan	[PXL1]
66	RVD	MIT Remote Virtual Disk Protocol	[MBG]
67	IPPC	Internet Pluribus Packet Core	[SHB]
68		any distributed file system	[IANA]
69	SAT-MON	SATNET Monitoring	[SHB]
70	VISA	VISA Protocol	[GXT1]
71	IPCV	Internet Packet Core Utility	[SHB]
72	CPNX	Computer Protocol Network Executive	[DXM2]
73	CPHB	Computer Protocol Heart Beat	[DXM2]
74	WSN	Wang Span Network	[VXD]
75	PVP	Packet Video Protocol	[SC3]
76	BR-SAT-MON	Backroom SATNET Monitoring	[SHB]
77	SUN-ND	SUN ND PROTOCOL-Temporary	[WM3]
78	WB-MON	WIDEBAND Monitoring	[SHB]
79	WB-EXPAK	WIDEBAND EXPAK	[SHB]
80	ISO-IP	ISO Internet Protocol	[MTR]

Maguire
maguire@it.kth.se

IP "Protocol" field (RFC 1700)
2006.03.13

Introduction 36 of 74
Internetworking/Internetteknik

Decimal	Keyword	Protocol	References
81	VMTP	VMTP	[DRC3]
82	SECURE-VMTP	SECURE-VMTP	[DRC3]
83	VINES	VINES	[BXH]
84	TTP	TTP	[JXS]
85	NSFNET-IGP	NSFNET-IGP	[HWB]
86	DGP	Dissimilar Gateway Protocol	[DGP,ML109]
87	TCF	TCF	[GAL5]
88	EIGRP	EIGRP	[CISCO,GXS]
89	OSPFIGP	OSPFIGP	[RFC1583,JTM4]
90	Sprite-RPC	Sprite RPC Protocol	[SPRITE,BXW]
91	LARP	Locus Address Resolution Protocol	[BXH]
92	MTP	Multicast Transport Protocol	[SXA]
93	AX.25	AX.25 Frames	[BK29]
94	IPIP	IP-within-IP Encapsulation Protocol	[JI6]
95	MICP	Mobile Internetworking Control Pro.	[JI6]
96	SCC-SP	Semaphore Communications Sec. Pro.	[HXH]
97	ETHERIP	Ethernet-within-IP Encapsulation	[RXH1]
98	ENCAP	Encapsulation Header	[RFC1241,RXB3]
99		any private encryption scheme	[IANA]
100	GMTP	GMTP	[RXB5]
101	IFMP	Ipsilon Flow Management Protocol	[Hinden]
102	PNNI	PNNI over IP	[Callon]
103	PIM	Protocol Independent Multicast	[Farinacci]

Maguire
maguire@it.kth.se

IP "Protocol" field (RFC 1700)
2006.03.13

Introduction 37 of 74
Internetworking/Internetteknik

Decimal	Keyword	Protocol	References
104	ARIS	ARIS	[Feldman]
105	SCPS	SCPS	[Durst]
106	QNX	QNX	[Hunter]
107	A/N	Active Networks	[Braden]
108	IPComp	IP Payload Compression Protocol	[RFC2393]
109	SNP	Sitara Networks Protocol	[Sridhar]
110	Compaq-Peer	Compaq Peer Protocol	[Volpe]
111	IPX-in-IP	IPX in IP	[Lee]
112	VRRP	Virtual Router Redundancy Protocol	[Hinden]
113	PGM	PGM Reliable Transport Protocol	[Speakman]
114		any 0-hop protocol	[IANA]
115	L2TP	Layer Two Tunneling Protocol	[Aboba]
116	DDX	D-II Data Exchange (DDX)	[Worley]
117	IATP	Interactive Agent Transfer Protocol	[Murphy]
118	STP	Schedule Transfer Protocol	[JMP]
119	SRP	SpectraLink Radio Protocol	[Hamilton]
120	UTI	UTI	[Lothberg]
121	SMP	Simple Message Protocol	[Ekblad]
122	SM	SM	[Crowcroft]
123	PTP	Performance Transparency Protocol	[Welzl]
124	ISIS	over IPv4	[Przygienda]
125	FIRE		[Partridge]
126	CRTP	Combat Radio Transport Protocol	[Sautter]

Maguire
maguire@it.kth.se

IP "Protocol" field (RFC 1700)
2006.03.13

Introduction 38 of 74
Internetworking/Internetteknik

Decimal	Keyword	Protocol	References
127	CRUDP	Combat Radio User Datagram	[Sautter]
128	SSCOPMCE		[Waber]
129	IPLT		[Hollbach]
130	SPS	Secure Packet Shield	[McIntosh]
131	PIPE	Private IP Encapsulation within IP	[Petri]
132	SCTP	Stream Control Transmission Protocol	[Stewart]
133	FC	Fibre Channel	[Rajagopal]
134	FRSVP-E2E-IGNORE		[RFC3175]
136	UDPLite		[RFC3828]
137	MPLS-in-IP		[RFC-ietf-mpls-in-ip-or-gre-08.txt]
138-252	Unassigned		[IANA]
253	Use for experimentation and testing		[RFC3692]
254	Use for experimentation and testing		[RFC3692]
255	Reserved		[IANA]

As of Jan. 2005, there are 4 fewer available protocol numbers than the course in 2003 and 41 fewer since the course in 1999.

Maguire
maguire@it.kth.se

IP "Protocol" field (RFC 1700)
2006.03.13

Introduction 39 of 74
Internetworking/Internetteknik

Basic communication mechanism: datagram

Properties of datagrams:

- Best effort

- Each message handled independently — global addressing.

- IP packets (datagrams) are forwarded according to the network address (which is in each datagram) by routers.

Maguire
maguire@it.kth.se

Basic communication mechanism: datagram
2006.03.13

Introduction 40 of 74
Internetworking/Internetteknik

Basic Ethernet + IP Software Architecture
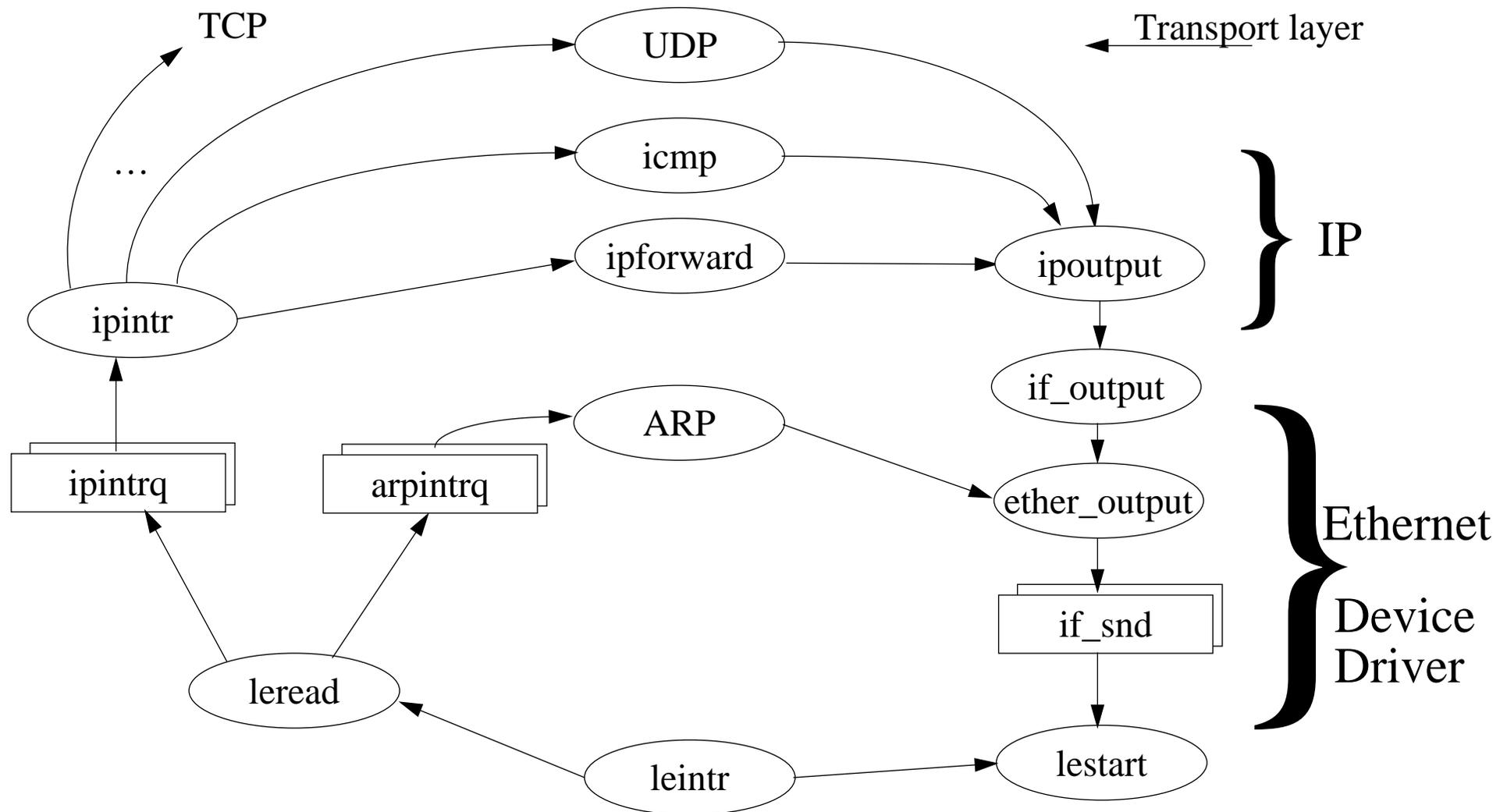
Figure 6: Basic Ethernet + IP Architecture - based on Stevens, TCP/IP Illustrated, Volume 2

Maguire
maguire@it.kth.se

Basic Ethernet + IP Software Architecture
2006.03.13

Introduction 41 of 74
Internetworking/Internetteknik

Common Used Simple Services

Name	TCP port	UDP port	RFC	Description
echo	7	7	862	server returns what the client sends
discard	9	9	863	server discards what the client sends
daytime	13	13	867	Server returns the time and date in a human readable format
chargen	19	19	864	TCP server sends a continual stream of character, until the connection is terminated by the client. UDP server sends a datagradm containing a random number of characters each time the client sends a datagram.
ftp-data	20			File Transfer Protocol (Data)
ftp	21			File Transfer Protocol (Control)
telnet	23			Virtual Terminal Protocol
smtp	25			Simple Mail Transfer Protocol
time	37	37	868	Server returns the time as a 32-bit binary number. This number is the time in seconds since 1 Jan. 1990, UTC

Maguire
maguire@it.kth.se

Common Used Simple Services
2006.03.13

Introduction 42 of 74
Internetworking/Internetteknik

Link Layer

Possible link layers include:

- Ethernet and IEEE 802.3 Encapsulation
  - with possible Trailer Encapsulation
- SLIP: Serial Line IP
- CSLIP: Compress SLIP
- PPP: Point to Point Protocol
- Loopback Interface
- Virtual Interface
- …
- carrier pigeons - CPIP (Carrier Pigeon Internet Protocol) April 1st 1990, RFC 1149 was written. A protocol for IP over avian carriers. Implementation (April 28 2001): *http://www.blug.linux.no/rfc1149/*
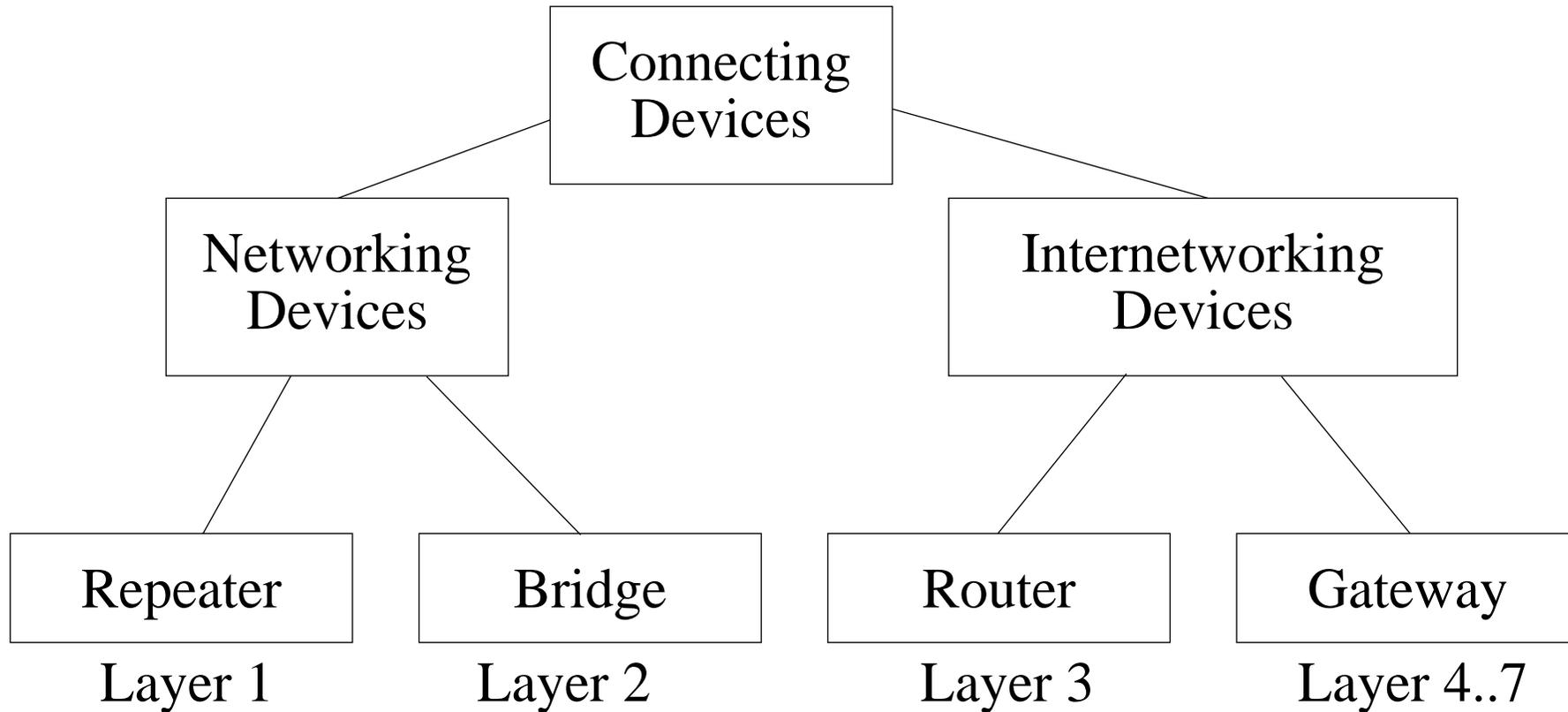
Some of the issues concerning links are:

- MTU and Path MTU
- Serial line throughput

Maguire
maguire@it.kth.se

Link Layer
2006.03.13

Introduction 43 of 74
Internetworking/Internetteknik

Simple Campus Network

(wireless) subnet$_1$ subnet$_2$ subnet$_3$ subnet$_4$ subnet$_5$

WAN

ISP's router

WAN or MAN   Firewall Router

Router

repeater layer 1

layer3 switch

bridge

hub

Collision domain   **layer 2 domain**   **Network domain**

Maguire
maguire@it.kth.se

Simple Campus Network
2006.03.13

Introduction 44 of 74
Internetworking/Internetteknik

Connecting Devices

```
                    ┌──────────────┐
                    │  Connecting  │
                    │   Devices    │
                    └──────────────┘
              ┌───────────┘      └───────────┐
    ┌──────────────┐              ┌──────────────────┐
    │  Networking  │              │ Internetworking  │
    │   Devices    │              │     Devices      │
    └──────────────┘              └──────────────────┘
      ┌──────┴──────┐              ┌──────┴──────┐
 ┌──────────┐  ┌──────────┐  ┌──────────┐  ┌──────────┐
 │ Repeater │  │  Bridge  │  │  Router  │  │ Gateway  │
 └──────────┘  └──────────┘  └──────────┘  └──────────┘
   Layer 1       Layer 2       Layer 3       Layer 4..7
```

- Ethernet hub = a multiport repeater
- Ethernet switch = a multiport bridge
- Layer 3 switch = combines functions of an ethernet switch and a router

Maguire
maguire@it.kth.se

Connecting Devices
2006.03.13

Introduction 45 of 74
Internetworking/Internetteknik

LAN Protocols

Data link Layer	LLC Sublayer
	MAC Sublayers
Physical Layer	

OSI Layers

| IEEE 802.2 | | | | |
| Ethernet | IEEE 802.3 | IEEE 802.4 Token Bus | IEEE 802.5 Token Ring | IEEE 802.11 WLAN | IEEE 802.15 PAN |

LAN specifications

Figure 7: Physical and Link layer protocols used for LANs

Maguire
maguire@it.kth.se

LAN Protocols
2006.03.13

Introduction 46 of 74
Internetworking/Internetteknik

Ethernet Encapsulation (RFC 894)
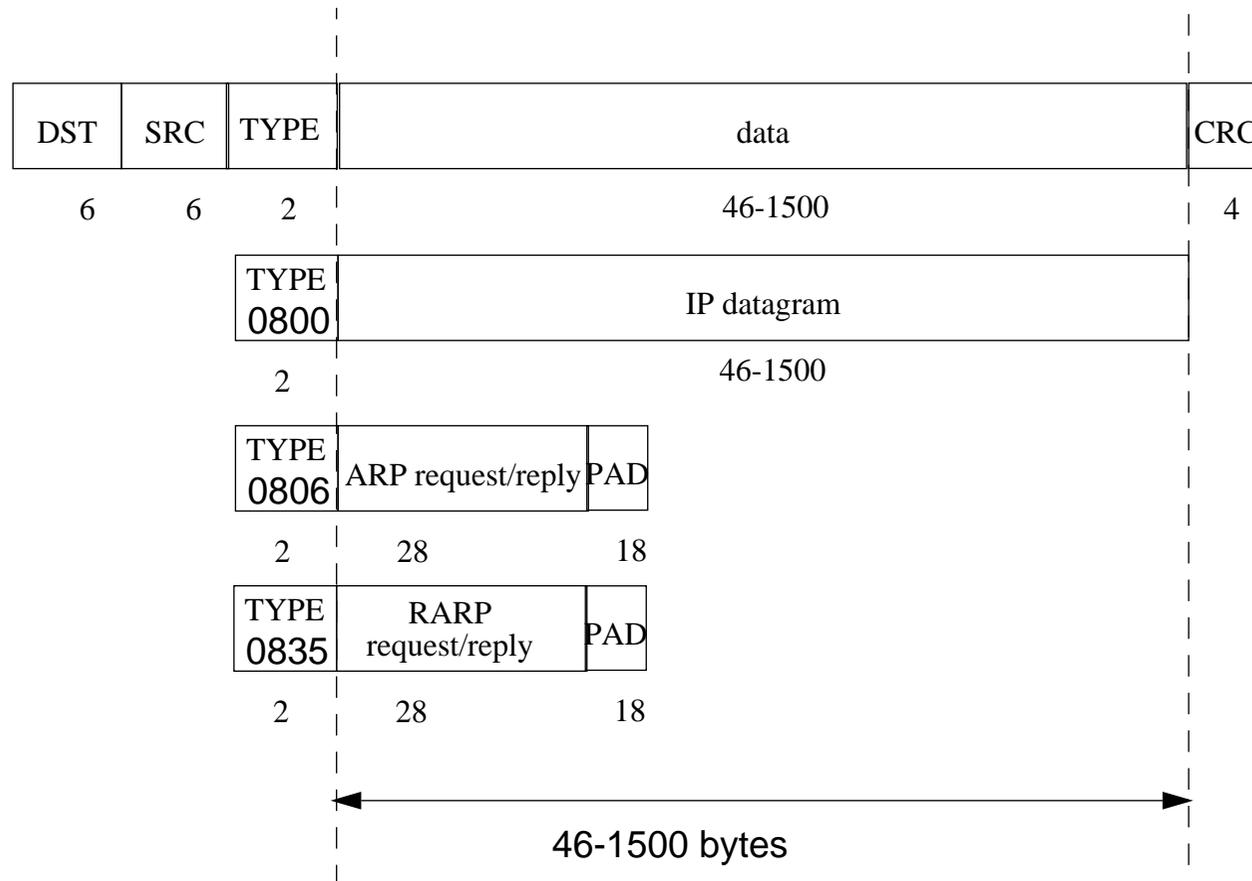
Figure 8: Ethernet encapsulation
(see Stevens, Volume 1, figure 2.1, pg. 23)
DST = Destination MAC Address, SRC = Source MAC Address (both are 48 bits in length);
TYPE = Frame Type; CRC = Cyclic Redundancy Check, i.e., checksum

Maguire
maguire@it.kth.se

Ethernet Encapsulation (RFC 894)
2006.03.13

Introduction 47 of 74
Internetworking/Internetteknik

IEEE 802.2/802.3 Encapsulation (RFC 1042)
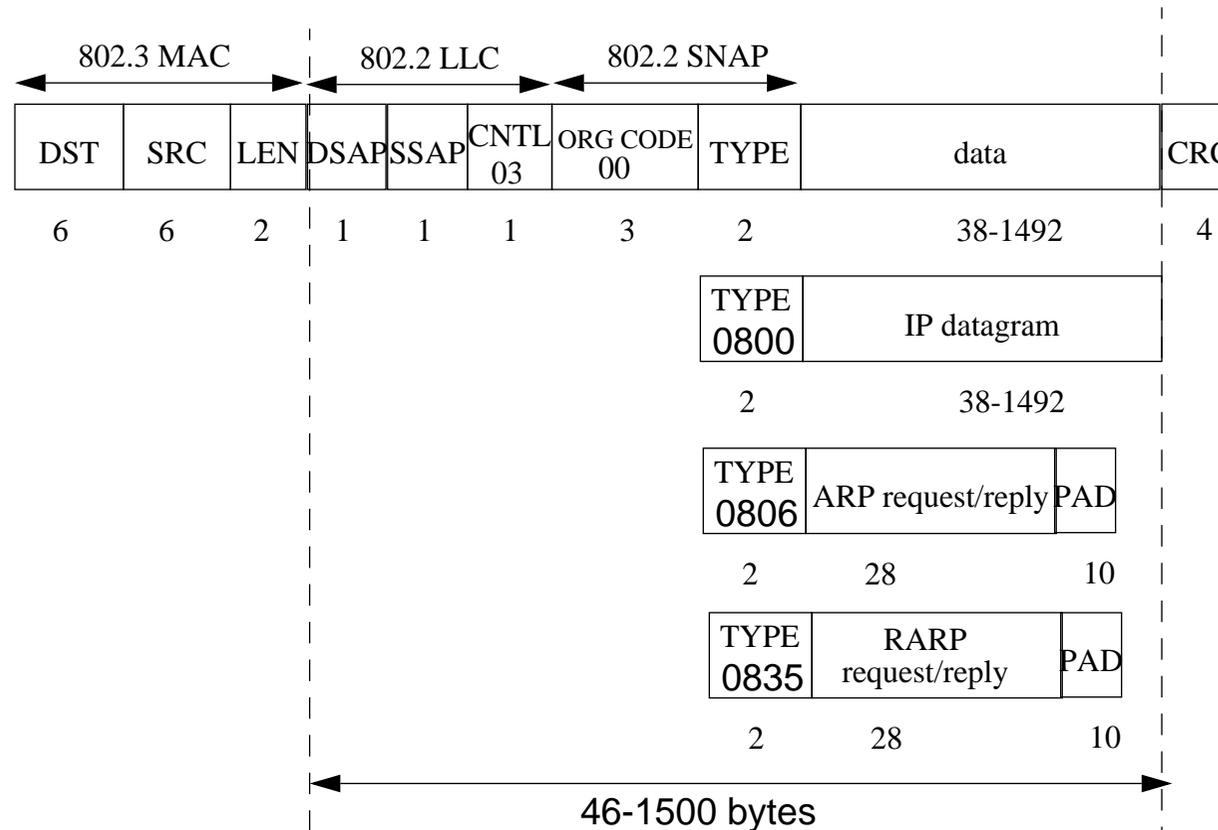
Figure 9: IEEE802.2/802.3 (see Stevens, Volume 1, figure 2.1, pg. 23)

DSAP ≡ Destination Service Access Point; SSAP ≡ Source Service Access Point;
SNAP ≡ Sub-Network Access Protocol; for other TYPE values see RFC1700.

Maguire
maguire@it.kth.se

IEEE 802.2/802.3 Encapsulation (RFC 1042)
2006.03.13

Introduction 48 of 74
Internetworking/Internetteknik

IEEE 802 Numbers of Interest

"… IEEE 802 Networks. These systems may use a Link Service Access Point (LSAP) field in much the same way the MILNET uses the "link" field. Further, there is an extension of the LSAP header called the Sub-Network Access Protocol (SNAP).

The IEEE likes to describe numbers in binary in bit transmission order, which is the opposite of the big-endian order used throughout the Internet protocol documentation."

Assignments from RFC1700

Link Service Access Point			Description	References
IEEE binary	Internet binary	decimal		
00000000	00000000	0	Null LSAP	[IEEE]
01000000	00000010	2	Individual LLC Sublayer Mgt	[IEEE]
11000000	00000011	3	Group LLC Sublayer Mgt	[IEEE]
00100000	00000100	4	SNA Path Control	[IEEE]
01100000	00000110	6	Reserved (DOD IP)	[RFC76]
01110000	00001110	14	PROWAY-LAN	[IEEE]
01110010	01001110	78	EIA-RS 511	[IEEE]
01111010	01011110	94	ISI IP	[JBP]
01110001	10001110	142	PROWAY-LAN	[IEEE]
01010101	10101010	170	SNAP	[IEEE]
01111111	11111110	254	ISO CLNS IS 8473	[RFC926]
11111111	11111111	255	Global DSAP	[IEEE]

Maguire

maguire@it.kth.se

IEEE 802 Numbers of Interest

2006.03.13

Introduction 49 of 74

Internetworking/Internetteknik
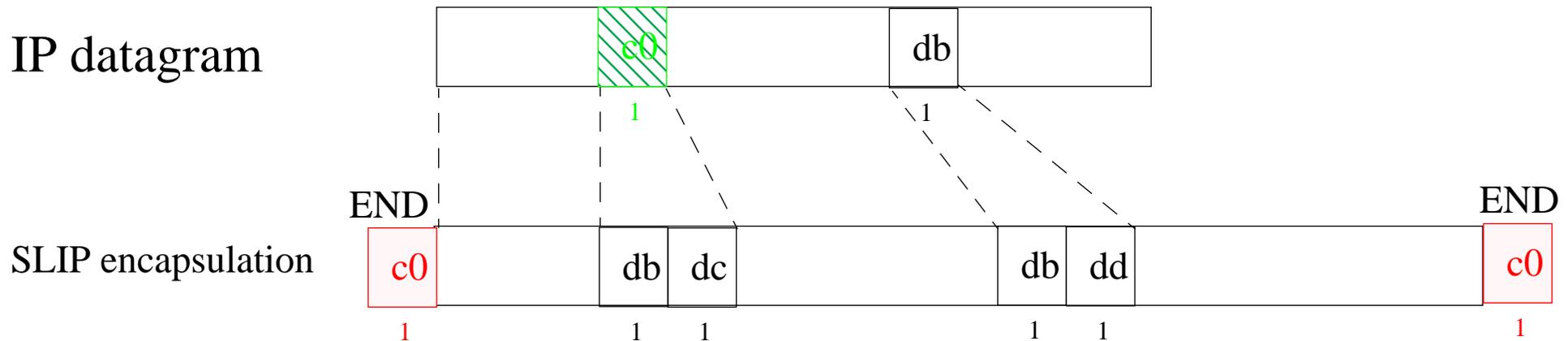
SLIP (RFC 1055)

Figure 10: SLIP Encapsulation (see Stevens, Volume 1, figure 2.2, pg. 25)

RFC 1055: *Nonstandard for transmission of IP datagrams over serial lines: SLIP*

SLIP uses character stuffing, SLIP ESC character $\equiv$ 0xdb

SLIP END character $\equiv$ 0xc0

- point to point link, $\Rightarrow$ no IP addresses need to be sent

- there is no TYPE field, $\Rightarrow$ you can only be sending IP, i.e., can't mix protocols

- there is no CHECKSUM, $\Rightarrow$ error detection has to be done by higher layers

Maguire
maguire@it.kth.se

SLIP (RFC 1055)
2006.03.13

Introduction 50 of 74
Internetworking/Internetteknik

SLIP Problems $\Rightarrow$ CSLIP $\equiv$ Compressed SLIP

- because many users running SLIP over lines at 19.2 kbits/s or slower
- lots of interactive traffic (telnet, rlogin, …) which uses TCP
  - many small packets
  - each of which needs a TCP header (20 bytes) + IP header (20 bytes) $\Rightarrow$ overhead 40 bytes
  - Send 1 user character requires sending a minimum of: 1 + 40 + END, i.e., 42 bytes
  - most of the header is **predictable**

CSLIP (RFC 1144: Compressing TCP/IP headers for low-speed serial links, by Van Jacobson) reduces the header to 3-5 bytes, by:

- trying to keep response time under 100-200ms
- keeping state about ~16 TCP connections at each end of the link
  - the 96-bit tuple <src address, dst address, src port, dst port> reduced to 4 bits
- many header fields rarely change - so don't transmit them
- some header fields change by a small amount - just send the delta
- no compression is attempted for UDP/IP
- a 5 byte compressed header on 100-200 bytes $\Rightarrow$ 95-98% line efficiency

Maguire
maguire@it.kth.se

SLIP Problems $\Rightarrow$ CSLIP $\equiv$ Compressed SLIP
2006.03.13

Introduction 51 of 74
Internetworking/Internetteknik

Robust Header Compression (rohc)

Header compression schemes that perform well over links with high error rates and long roundtrip times.

http://www.ietf.org/html.charters/rohc-charter.html

Maguire
maguire@it.kth.se

Robust Header Compression (rohc)
2006.03.13

Introduction 52 of 74
Internetworking/Internetteknik

PPP: Point to Point Protocol (RFC 1331, 1332)

PPP corrects the deficiencies in SLIP. PPP consists of:

- encapsulation for either async or synchronous links,
  - HDLC (see RFC 1549)
  - X.25 (see RFC 1598)
  - ISDN (see RFC 1618)
  - SONET/SDH (see RFC 1619)
- Link Control Protocol
  - establish, configure, and test data-links [includes option negociation]
  - authentication (see RFC 1334)
- Family of Network Control Protocols (NCPs) - specific to different network protocols, currently:
  - IP (see RFC 1332)
  - DECnet (see RFC 1376)
  - OSI network layer (see RFC 1377)
  - AppleTalk (see RFC 1378)
  - XNS (see RFC 1764)

See "PPP Design, Implementation, and Debugging", by James D. Carlson, Second edition, Addison-Wesley,2000, ISBN 0-201-70053-0.

Maguire
maguire@it.kth.se

PPP: Point to Point Protocol (RFC 1331, 1332)
2006.03.13

Introduction 53 of 74
Internetworking/Internetteknik

PPP frames

FLAG 7E	ADDR FF	CNTL 03	protocol	data	CRC	FLAG 7E
1	1	1	2	upto 1500 bytes	2	1

protocol 0021	IP datagram
2	

protocol C021	Link control data
2	

protocol 8021	network control data
2	
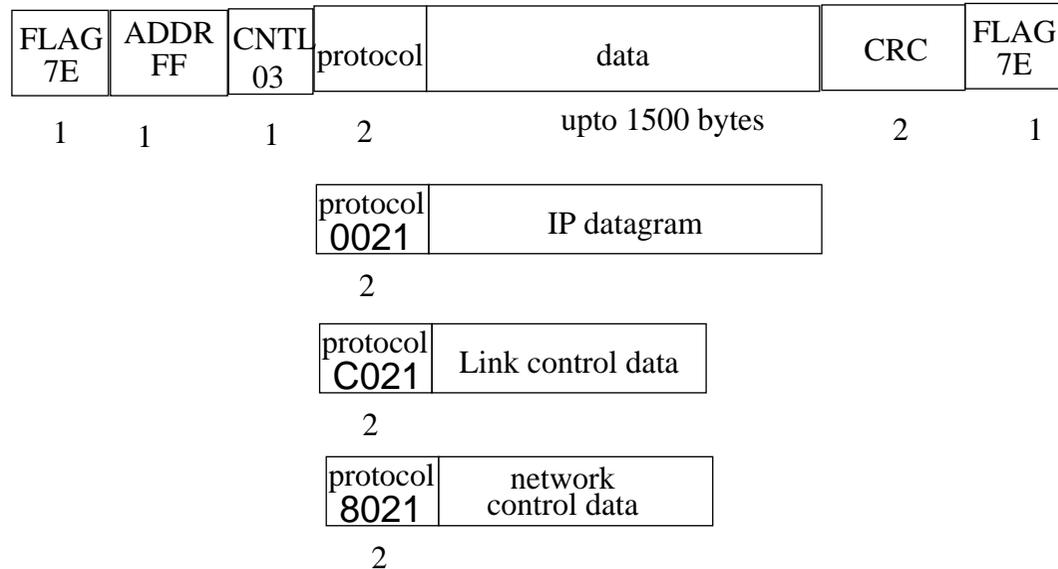
Figure 11: Format of PPP frame (see Stevens, Volume 1, figure 2.3, pg. 26)

- The protocol field behaves like the Ethernet TYPE field.
- CRC can be used to detect errors in the frame.
- Either character or bit stuffing is done depending on the link.
- you can negotiate away the CNTL and ADDRESS fields, and reduce the protocol field to 1 byte $\Rightarrow$ minimum overhead of 3 bytes
- Van Jacobson header compression for IP and TCP

Maguire
maguire@it.kth.se

PPP frames
2006.03.13

Introduction 54 of 74
Internetworking/Internetteknik

PPP summary

- support for multiple protocols on a link
- CRC check on every frame
- dynamic negociation of IP address of each end
- header compression (similar to CSLIP)
- link control with facilities for negotiating lots of data-link options

All at a price averaging 3 bytes of overhead per frame.

Maguire
maguire@it.kth.se

PPP summary
2006.03.13

Introduction 55 of 74
Internetworking/Internetteknik

Loopback interface
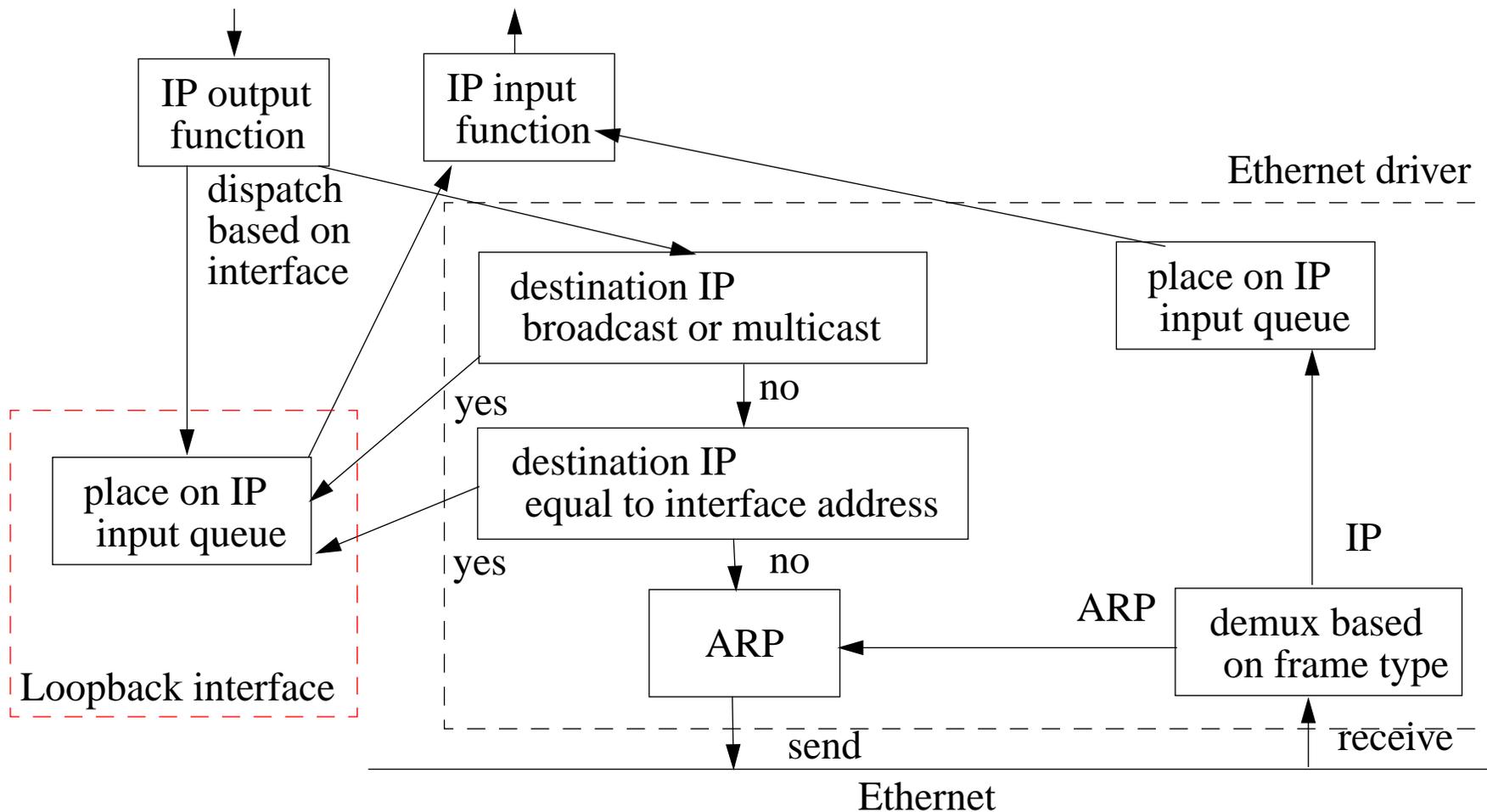
Figure 12: Processing of IP Datagrams (adapted from Stevens, Volume 1, figure 2.4, pg. 28)

Maguire
maguire@it.kth.se

Loopback interface
2006.03.13

Introduction 56 of 74
Internetworking/Internetteknik

Loopback interface summary

- loopback address $\equiv$ 127.0.0.1 generally called "localhost"
- all broadcasts and multicasts get sent to the loopback - because the sender gets a copy too!
- everything sent to the host's **own** IP address is sent to the loopback interface

Maguire
maguire@it.kth.se

Loopback interface summary
2006.03.13

Introduction 57 of 74
Internetworking/Internetteknik

Virtual Interface (VIF)
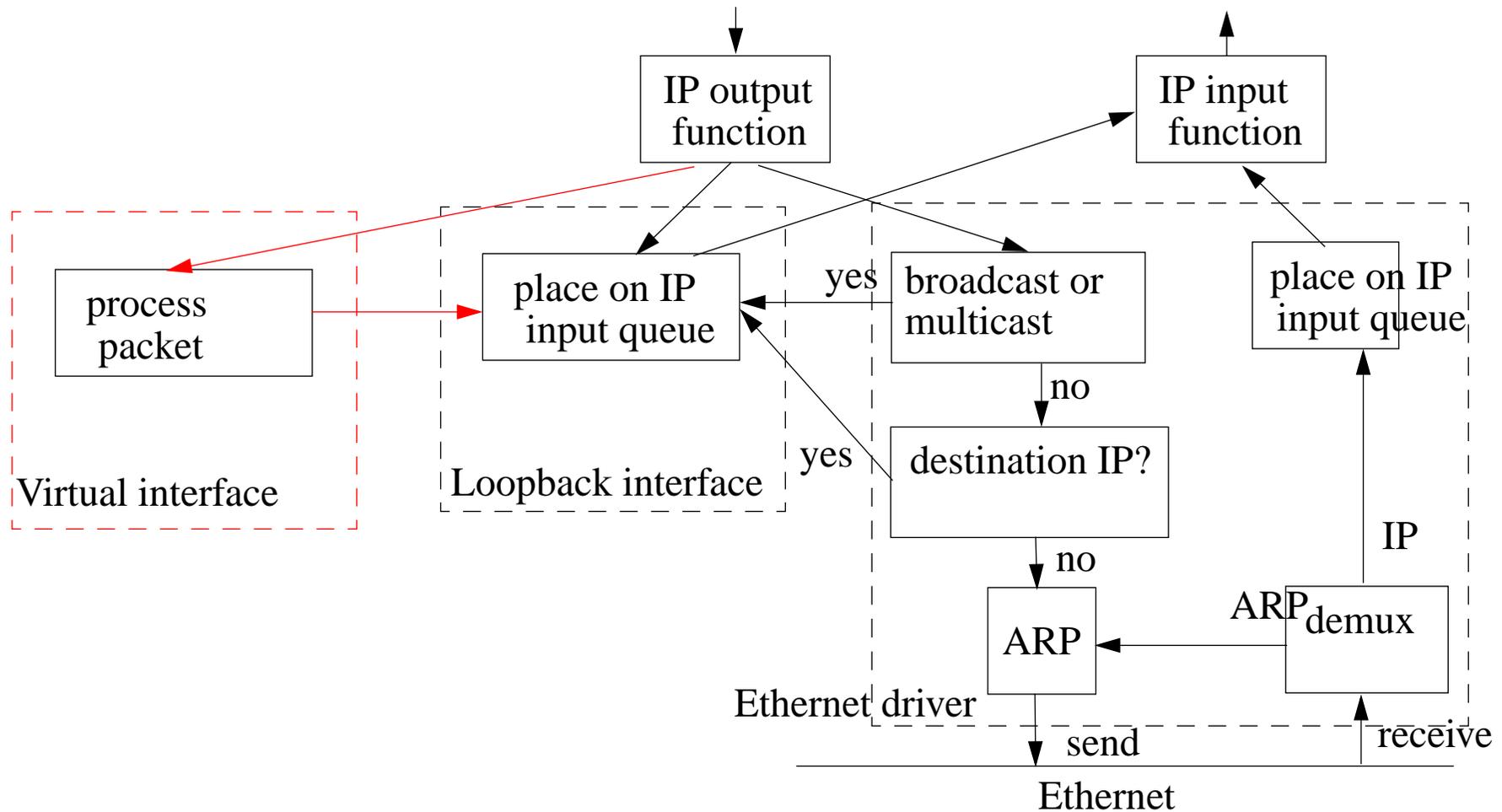
Figure 13: Processing of network packets via a Virtual Interface

Maguire
maguire@it.kth.se

Virtual Interface (VIF)
2006.03.13

Introduction 58 of 74
Internetworking/Internetteknik

Using VIF for tunneling
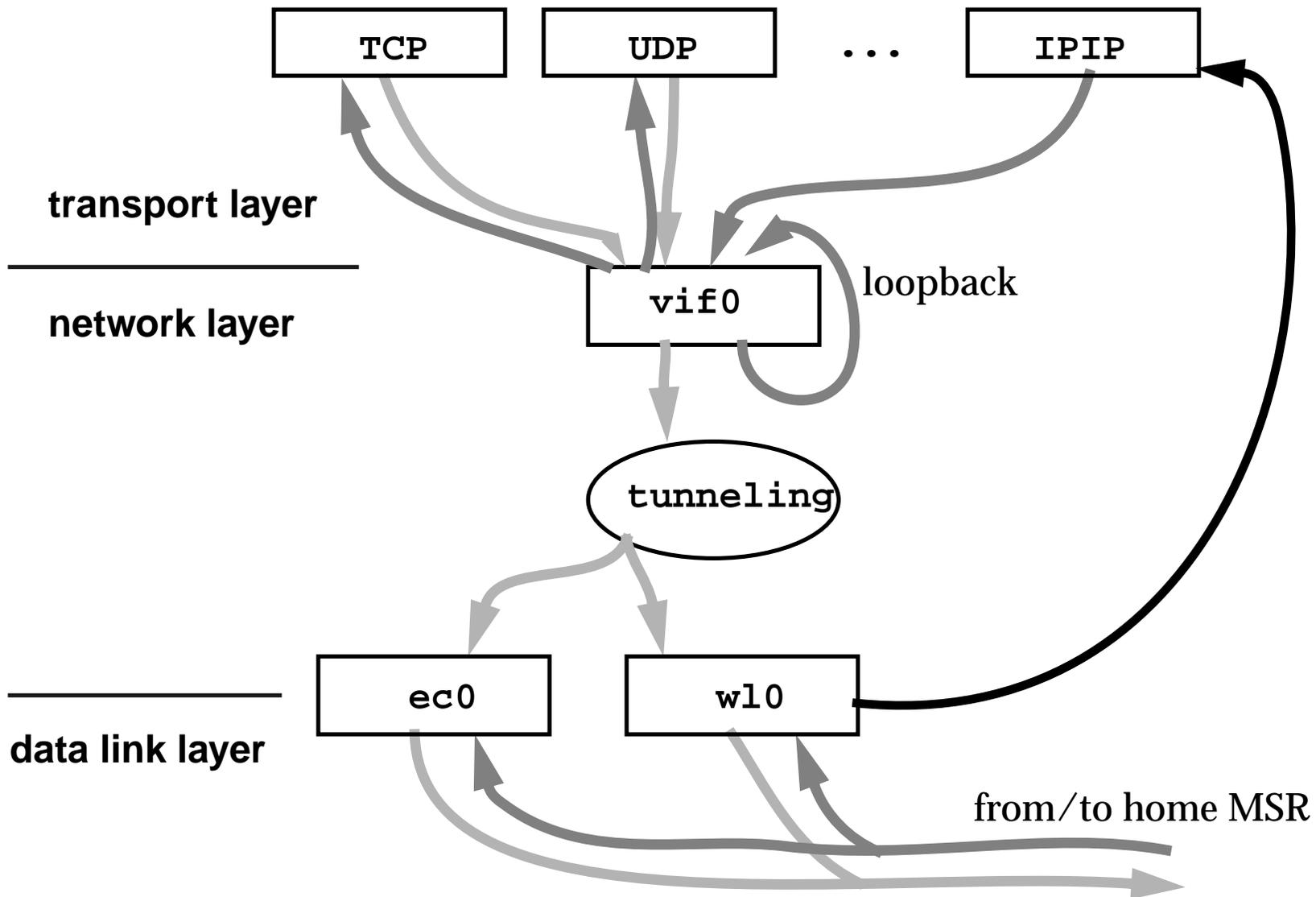
Figure 14: Using a Virtual Interface for Tunneling (IP in IP) adapted from John Ioannidis's thesis

Maguire
maguire@it.kth.se

Using VIF for tunneling
2006.03.13

Introduction 59 of 74
Internetworking/Internetteknik

IP addresses

Address types

- Unicast = one-to-one
- Multicast = one-to-many
- Broadcast = one-to-all

32 bit address divided into two parts:

NetID	Host ID

Figure 15: IP address format

Note that although we refer to it as the Host ID part of the address, it is really the address of an interface.

Dotted decimal notation: write each byte as a decimal number, separate each of these with a "." i.e., 10000010 11101101 00100000 00110011 $\Rightarrow$ 130.237.32.51 or in hexadecimal as: 0x82ED2033

Maguire
maguire@it.kth.se

IP addresses
2006.03.13

Introduction 60 of 74
Internetworking/Internetteknik

Classful addressing

Classically the address range was divided into classes:

Class	NetID		Range (dotted decimal notation)	host ID
A	0	+ 7- bit NetID	0.0.0.0 to **127**.255.255.255	24 bits of host ID
B	1 0	+ 14-bit NetID	**128**.0.0.0 to **191**.255.255.255	16 bits of host ID
C	1 1 0	+ 21-bit NetID	**192**.0.0.0 to **223**.255.255.255	8 bits of host ID
D	1 1 1 0		**224**.0.0.0 to **239**.255.255.255	28 bits of Multicast address
E	1 1 1 1 0		**240**.0.0.0 to **247**.255.255.255	Reserved for future use

- Globally addressable IP addresses must be unique
  - later in the course we will see how NATs affect this
- addresses roughly $2^7 * 2^{24} + 2^{14} * 2^{16} + 2^{21} * 2^8 = 3,758,096,384$ interfaces (**not** the number of hosts)
- in 1983 this seemed like a lot of addresses
- problems with the size of the blocks $\Rightarrow$ lots of wasted addresses
  - lead to classless addressing!

Maguire
maguire@it.kth.se

Classful addressing
2006.03.13

Introduction 61 of 74
Internetworking/Internetteknik

Classless addressing: Subnetting IP networks

Often we want to "subnet" - i.e., divide the network up into multiple networks:

NetID	SubnetID	Host
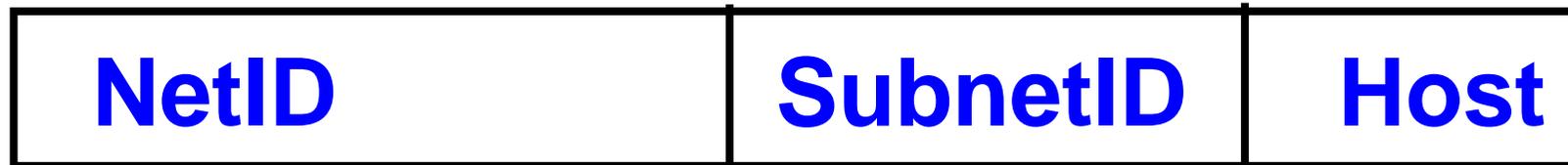
Figure 16: IP addresses and subnetting

Although the Subnet field is shown as a field which is separate from the Host field, it could actually be divided on a bit by bit basis; this is done by a **Subnet Mask.**

A common practice to avoid wasting large amounts of address space is to use Classless Interdomain Routing (CIDR) also called "supernetting" {see §10.8 of Steven's Vol. 1 and RFCs 1518 and 1519}.

Maguire
maguire@it.kth.se

Classless addressing: Subnetting IP networks
2006.03.13

Introduction 62 of 74
Internetworking/Internetteknik

Special Case IP Addresses

IP Address			Can appear as		Description
net ID	**subnet ID**	**host ID**	**source?**	**destination**	
0		0	OK	never	**this** host on **this** net
0		**hostid**	OK	never	specified host on **this** net
127		**any**	OK	OK	loopback address
-1		-1	never	OK	limited broadcast (never forwarded)
netid		-1	never	OK	net-directed broadcast to **netid**
netid	**subnetid**	-1	never	OK	subnet-directed broadcast to **netid**, **subnetid**
netid	-1	-1	never	OK	all-subnets-directed broadcast to **netid**

Thus for every subnet - the zero host ID address refers to **this net** and the all ones host ID is a **subnet broadcast** address; this uses up two addresses from every subnet's address range.

Maguire
maguire@it.kth.se
Special Case IP Addresses
2006.03.13
Introduction 63 of 74
Internetworking/Internetteknik

Subnet mask

32 bit value with a 1 for NetID + subnetID, 0 for HostID

16 bits	8 bits	8 bits
NetID	**SubnetID**	**HostID**
1111 1111 1111 1111	1111 1111	0000 0000
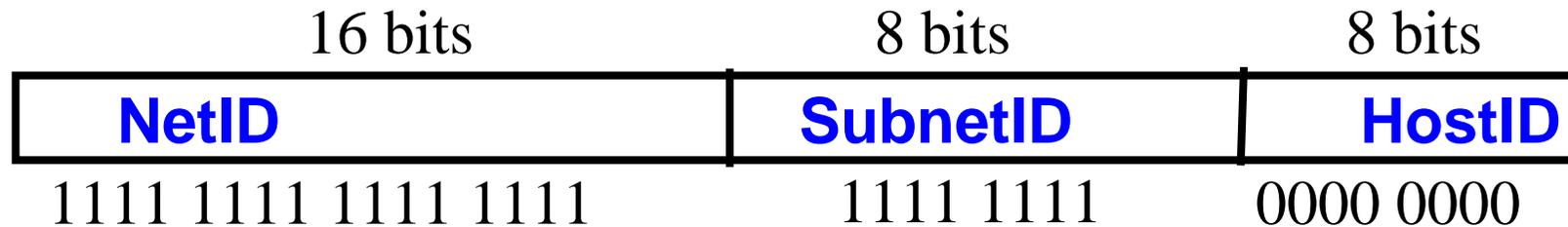
Figure 17: Class B address with a 8 bit subnet address

2 different class B subnet arrangements

16 bits	8 bits	8 bits	
NetID	**SubnetID**	**HostID**	0xffffff00
1111 1111 1111 1111	1111 1111	0000 0000	255.255.255.0 /24

16 bits	10 bits	6 bits	
NetID	**SubnetID**	**HostID**	0xffffffc0
1111 1111 1111 1111	1111 1111	1100 0000	255.255.255.192 /26

Maguire
maguire@it.kth.se

Subnet mask
2006.03.13

Introduction 64 of 74
Internetworking/Internetteknik
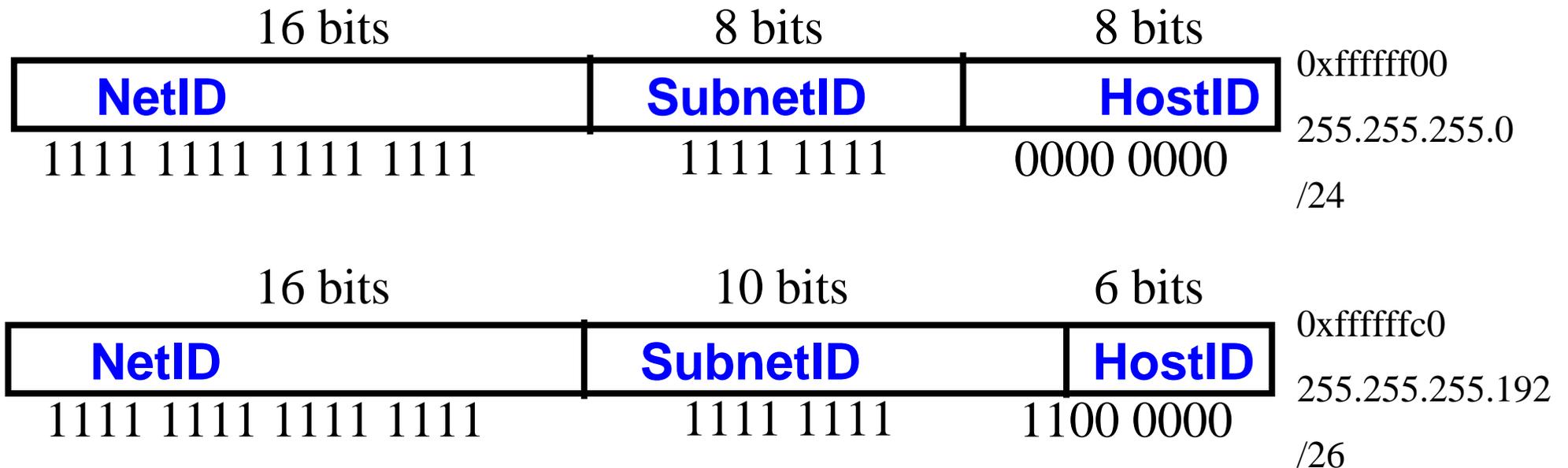
Classless Inter-Domain Routing (CIDR)

Slash notation

Length (CIDR)	Address mask	Notes	Length (CIDR)	Address mask	Notes
/0	0.0.0.0	All 0's ≡ no mask	/8	255.0.0.0	≡ Class A
/1	128.0.0.0		/9	255.128.0.0	
/2	192.0.0.0		/10	255.192.0.0	
/3	224.0.0.0		/11	255.224.0.0	
/4	240.0.0.0		/12	255.240.0.0	
/5	248.0.0.0		/13	255.248.0.0	
/6	252.0.0.0		/14	255.252.0.0	
/7	254.0.0.0		/15	255.254.0.0	

Maguire
maguire@it.kth.se

Classless Inter-Domain Routing (CIDR)
2006.03.13

Introduction 65 of 74
Internetworking/Internetteknik

Slash notation continued

Length (CIDR)	Address mask	Notes	Length (CIDR)	Address mask	Notes
/16	255.255.0.0	≡ Class B	/24	255.255.255.0	≡ Class C
/17	255.255.128.0		/25	255.255.255.128	
/18	255.255.192.0		/26	255.255.255.192	
/19	255.255.224.0		/27	255.255.255.224	
/20	255.255.240.0		/28	255.255.255.240	
/21	255.255.248.0		/29	255.255.255.248	
/22	255.255.252.0		/30	255.255.255.252	
/23	255.255.254.0		/31	255.255.255.254	
			/32	255.255.255.255	All 1's (host specific mask)

Maguire
maguire@it.kth.se

Classless Inter-Domain Routing (CIDR)
2006.03.13

Introduction 66 of 74
Internetworking/Internetteknik

IP address assignments

Internet Service Providers (ISPs) should contact their upstream registry or their appropriate Regional Internet Registries (RIR) at one of the following addresses:

Region	
APNIC (Asia-Pacific Network Information Center)	`http://www.apnic.net`
ARIN (American Registry for Internet Numbers)	`http://www.arin.net`
RIPE NCC (Reseau IP Europeens)	`http://www.ripe.net`

Private addresses - these IP addresses are for strictly **private** use:

Class	Netids	block
A	10.	1
B	172.16 to 172.31	16
C	192.168.0 to 192.168.255	256

Maguire
maguire@it.kth.se
IP address assignments
2006.03.13
Introduction 67 of 74
Internetworking/Internetteknik

Problems with the dual functions of IP addresses

Unfortunatelly an IP address has dual functions:

- Network ID portion indicates a location in the network
  - i.e., the network ID binds the address to a location in the network topology
  - CIDR and hierarchical address prefixes - allow for recursive subdivision of the topology

- Host ID portion identifies an interface - often used as a node identifier
  - Unfortunately network connections are bound to these identifiers
  - Specifically TCP/UDP sockets are identified by the endpoint IP address (and port numbers)
  - DNS returns one or more addresses for new connections

$\Rightarrow$ This is bad for mobility and multi-homing (see textbook figure 4.12 on pg. 95)
  - If a host changes its point of network attachment it must change its identity
    – Later we will see how Mobile IP addresses this problem
  - Host with multiple interfaces are limited in how they can use them
    – Later we will see how SCTP addresses part of this problem

The result has been that multiple and dynamic addresses are difficult to handle and lead to a number of efforts to rethink how addresses are used.

Maguire
maguire@it.kth.se

Problems with the dual functions of IP addresses
2006.03.13

Introduction 68 of 74
Internetworking/Internetteknik

ifconfig, route and netstat Commands

- ifconfig: to configure interface.
- route: to update routing table.
- netstat: to get interface and routing information.

For example: to configure interface, add an network and add a gateway:

    root# ifconfig eth0 192.71.20.115 netmask 255.255.255.0 up

    root# route add -net 192. 71.20.0 netmask 255.255.255.0 eth0

    root# route add default gw 192.71.20.1 eth0

We will discuss these commands in more detail in following lectures and in the recitations.

Maguire
maguire@it.kth.se

ifconfig, route and netstat Commands
2006.03.13

Introduction 69 of 74
Internetworking/Internetteknik

Standardization Organizations

The most relevant to the Internet are:

- ## Internet Society (ISOC)
    - Internet Engineering Task Force (IETF)
- ## World-wide-web consortium (W3C)
- ## International Standards Organization (ISO)
- ## International Telecommunications Union - Telecommunication Standards Sector (ITU-T)
- ## Institute of Electrical and Electronics Engineers (IEEE)
- ## …

Read in the textbook sections 1.4 and 1.5.

Maguire
maguire@it.kth.se

Standardization Organizations
2006.03.13

Introduction 70 of 74
Internetworking/Internetteknik

Summary

- Course Introduction
- Internet Basics
  - Multiplexing and demultiplexing
  - Dataggrams
- Link Layer Protocols for the Internet
  - Ethernet
  - SLIP, PPP
- IP: Internet Protocol
  - IP addressing
  - Subnetting

Maguire
maguire@it.kth.se

Summary
2006.03.13

Introduction 71 of 74
Internetworking/Internetteknik

W. Richard Stevens

- Born in Luanshya, Northern Rhodesia (now Zambia) in 1951
- Died on September 1, 1999
- He studied Aerospace Engineering, Systems Engineering (image processing major, physiology minor)
- flight instructor and programmer
- His many books helped many people to understand and use TCP/IP
  - UNIX Network Programming, Prentice Hall, 1990.
  - Advanced Programming in the UNIX Environment, Addison-Wesley, 1992.
  - TCP/IP Illustrated, Volume 1: The Protocols, Addison-Wesley, 1994.
  - TCP/IP Illustrated, Volume 2: The Implementation, Addison-Wesley, 1995.
  - TCP/IP Illustrated, Volume 3: TCP for Transactions, HTTP, NNTP, and the UNIX Domain Protocols, Addison-Wesley, 1996.
  - UNIX Network Programming, Volume 1, Second Edition: Networking APIs: Sockets and XTI, Prentice Hall, 1998.
  - UNIX Network Programming, Volume 2, Second Edition: Interprocess Communications, Prentice Hall, 1999.

Maguire
maguire@it.kth.se

W. Richard Stevens
2006.03.13

Introduction 72 of 74
Internetworking/Internetteknik

References

[1]   Barry M. Leiner, Vinton G. Cerf, David D. Clark, Robert E. Kahn, Leonard Kleinrock, Daniel C. Lynch, Jon Postel, Larry G. Roberts, and Stephen Wolff, "A Brief History of the Internet", On The Internet, May/June 1997

   `http://www.isoc.org/oti/articles/0597/leiner.html`

[2]   R. Kahn, Communications Principles for Operating Systems. Internal BBN memorandum, Jan. 1972.

[3]   V. Cerf and R. Kahn, "A protocol for packet network interconnection", IEEE Transactions on Communications Technology, Vol. COM-22, Number 5, May 1974, pp. 627-641.

   `http://global.mci.com/us/enterprise/insight/cerfs_up/technical_writings/protocol_paper/`

[4]   Jerome H. Saltzer, David P. Reed, David D. Clark, "End-To-End Arguments In System Design" In ACM Transactions on Computer Systems, V2, #4, Nov. 1984, pages 277-288

   `http://citeseer.ist.psu.edu/saltzer84endtoend.html`

Maguire
maguire@it.kth.se

References
2006.03.13

Introduction 73 of 74
Internetworking/Internetteknik

[5]   David D. Clark and Marjory S. Blumenthal, "Rethinking the Design of the Internet: The end to end arguments vs. the brave new world", In ACM Transactions on Internet Technology, Vol 1, No 1, August 2001, pp 70-109.

http://www.ana.lcs.mit.edu/papers/PDF/Rethinking_2001.pdf

[6]   D. Clark, J. Wroclawski, K. Sollins, and R. Braden, "Tussle in Cyberspace: Defining Tomorrow's Internet", Proceedings of Sigcomm 2002.

http://www.acm.org/sigs/sigcomm/sigcomm2002/papers/tussle.pdf

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 2: IP Basics: Routing, ARP, and RARP

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapters 6 - 8

KTH Information and Communication Technology

IP Basics Outline

- IP Routing: Delivery and Routing of IP packets
- Address Resolution: ARP and RARP

Maguire
maguire@it.kth.se

IP Basics Outline
2006.03.13

IP Basics: Routing, ARP, and RARP 76 of 121
Internetworking/Internetteknik

Connection-oriented vs Connectionless

- Connection-Oriented Services
  - Network layer first establishes a connection between a source and a destination
  - Packets are sent along this connection
  - Route is decided **once** at the time the connection is established
  - Routers/switches in connection-oriented networks are **stateful**

- Connectionless Services
  - Network layer can process each packet **independently**
  - A route lookup is performed for **each** packet
  - IP is connectionless
  - IP routers are **stateless**

Of course reality is (much) more complex, to gain performance IP routers dynamically create state (in caches) as there is frequently **correlation** between packets (i.e., if you just did a route lookup for destination B, there is a non-zero probability that another packet which will arrive shortly might also be headed to destination B).

Maguire
maguire@it.kth.se
Connection-oriented vs Connectionless   IP Basics: Routing, ARP, and RARP 77 of
2006.03.13
Internetworking/Internetteknik

Routing

The internet protocols are based on moving packets from a source to a destination with each hop making a routing decision.

Two components to routing:

- packet forwarding - Routing Mechanism: search the routing table and decide which interface to send a packet out.
  - A matching host address? If no,
  - A matching network address? (using longest match) If no,
  - Default entry.
- computing routes - Routing Policy: rules that decide which routes should be added into the routing table.

Traditionally most of the complexity was in the later (i.e., computing routes) while packet forwarding was very straight forward -- this is no longer true due to QoS.

Routers vs. hosts   -- a node can be both

- Routers forward IP packets
- Hosts generate or sink IP packets

Maguire
maguire@it.kth.se

Routing
2006.03.13

IP Basics: Routing, ARP, and RARP 78 of 121
Internetworking/Internetteknik

Direct vs. indirect Delivery

- ## Direct delivery
  - The final destination is (directly) connected to the same physical network as the sender
  - IP destination address and local interface have the same netmask
  - Map destination IP address to destination physical address via ARP
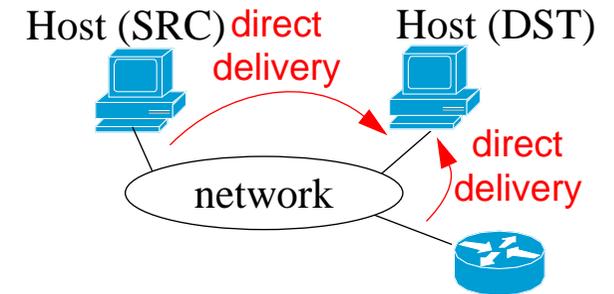
- ## Indirect delivery
  - From router to router (note: the last delivery is always direct!)
  - Destination address is used for a routing lookup in a routing table: Routing
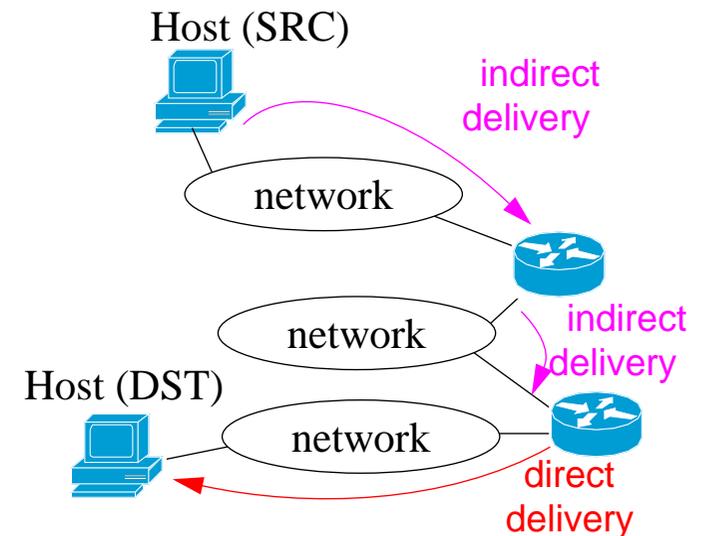
SRC=Source
DST = Destination

Host (SRC) direct delivery   Host (DST)

direct delivery

network

Host (SRC)

indirect delivery

network

network

indirect delivery

Host (DST)

network

direct delivery

Maguire
maguire@it.kth.se

Direct vs. indirect Delivery
2006.03.13

IP Basics: Routing, ARP, and RARP 79 of 121
Internetworking/Internetteknik

Forwarding

- Next-Hop method - routing table holds only the address of the next hop

Routing table for A	
Destination	Route
Host B	R1

Routing table for R1	
Destination	Route
Host B	R2

Routing table for R2	
Destination	Route
Host B	
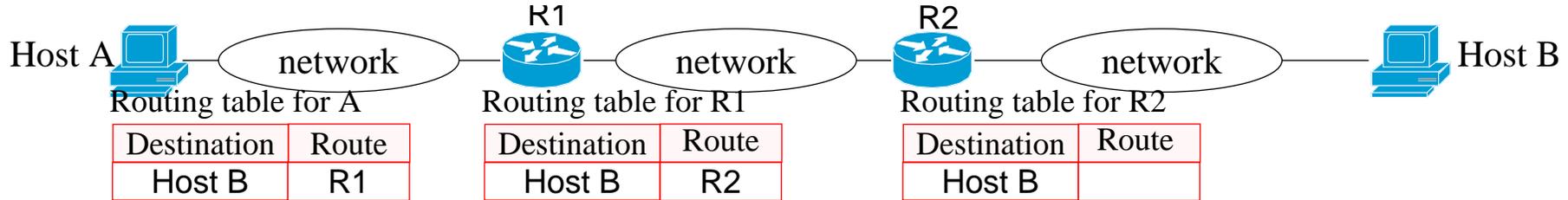
- Network-specific method - routing table entries are for networks

Routing table for A	
Destination	Route
network2	R1
network3	R1

Routing table for R1	
Destination	Route
network3	R2

Routing table for R2	
Destination	Route
network1	R2
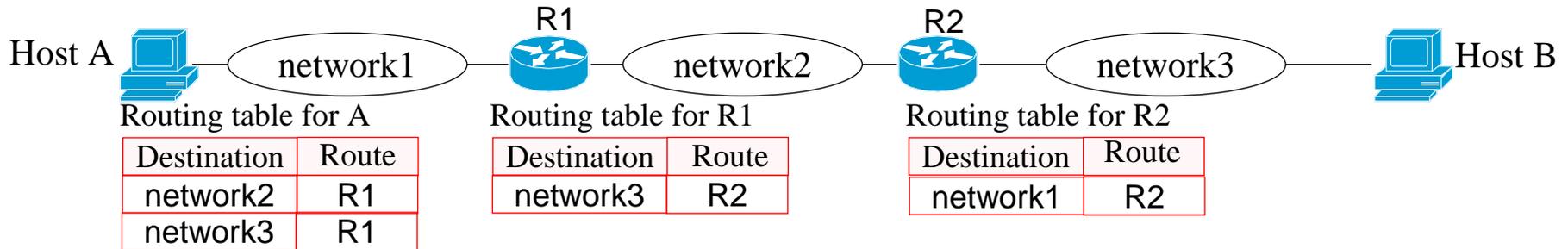
- Host-specific method - per host routes

- Default method - specifies a default route (normally network address 0.0.0.0)

Routing table for A	
Destination	
network2	R1
default	R2
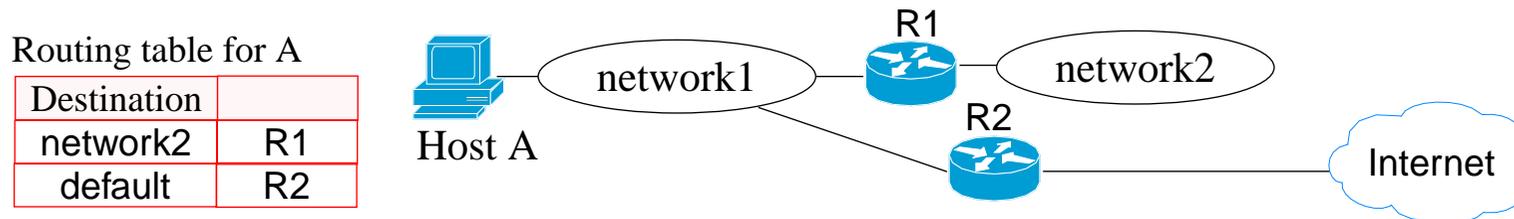
See textbook sections 6.2 for more examples.

Maguire
maguire@it.kth.se

Forwarding
2006.03.13

IP Basics: Routing, ARP, and RARP 80 of 121
Internetworking/Internetteknik

Processing

Maguire
maguire@it.kth.se

Processing
2006.03.13

IP Basics: Routing, ARP, and RARP 81 of 121
Internetworking/Internetteknik
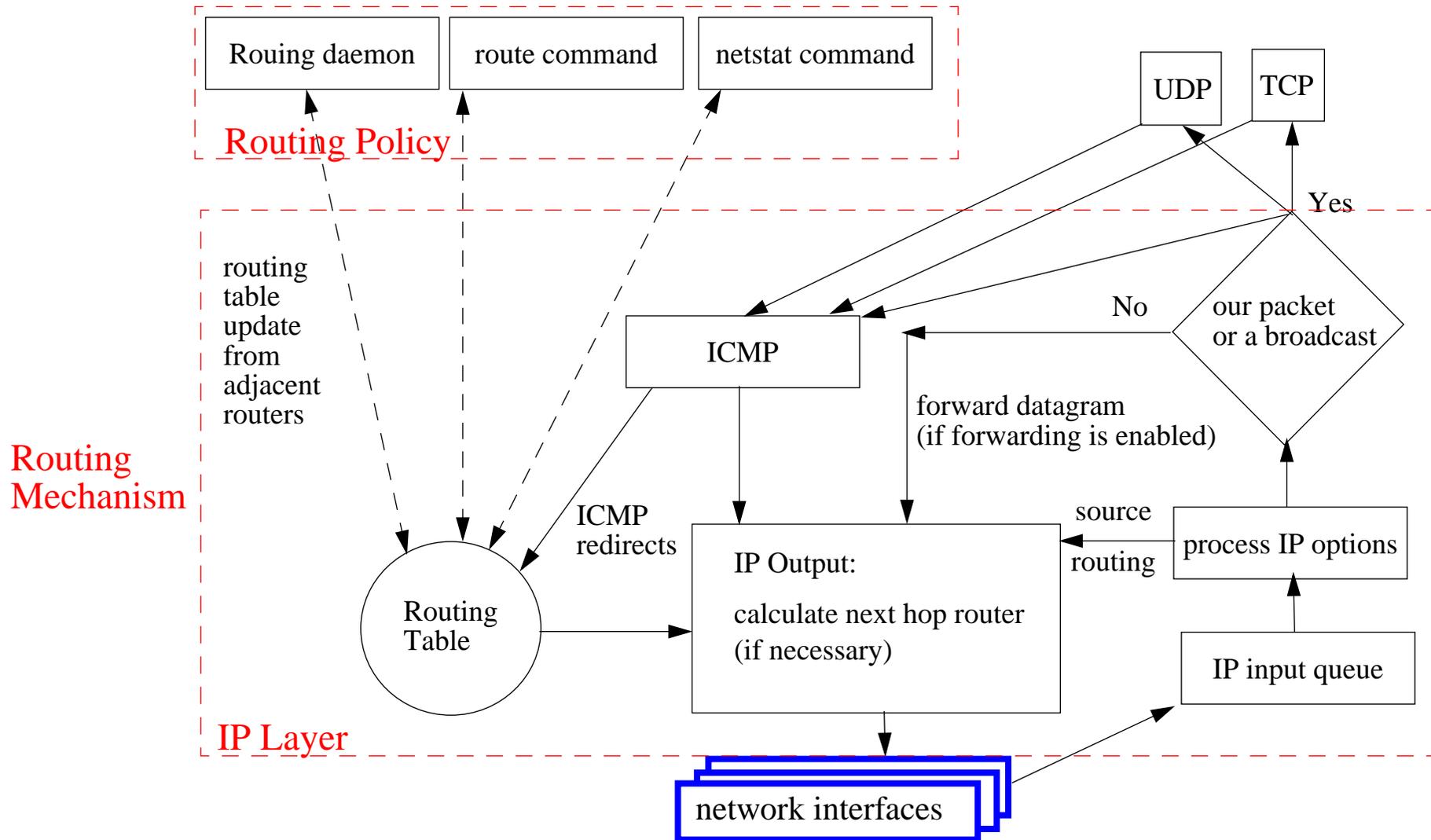
Forwarding module

A simplified view of forwarding using classful address without subnetting:
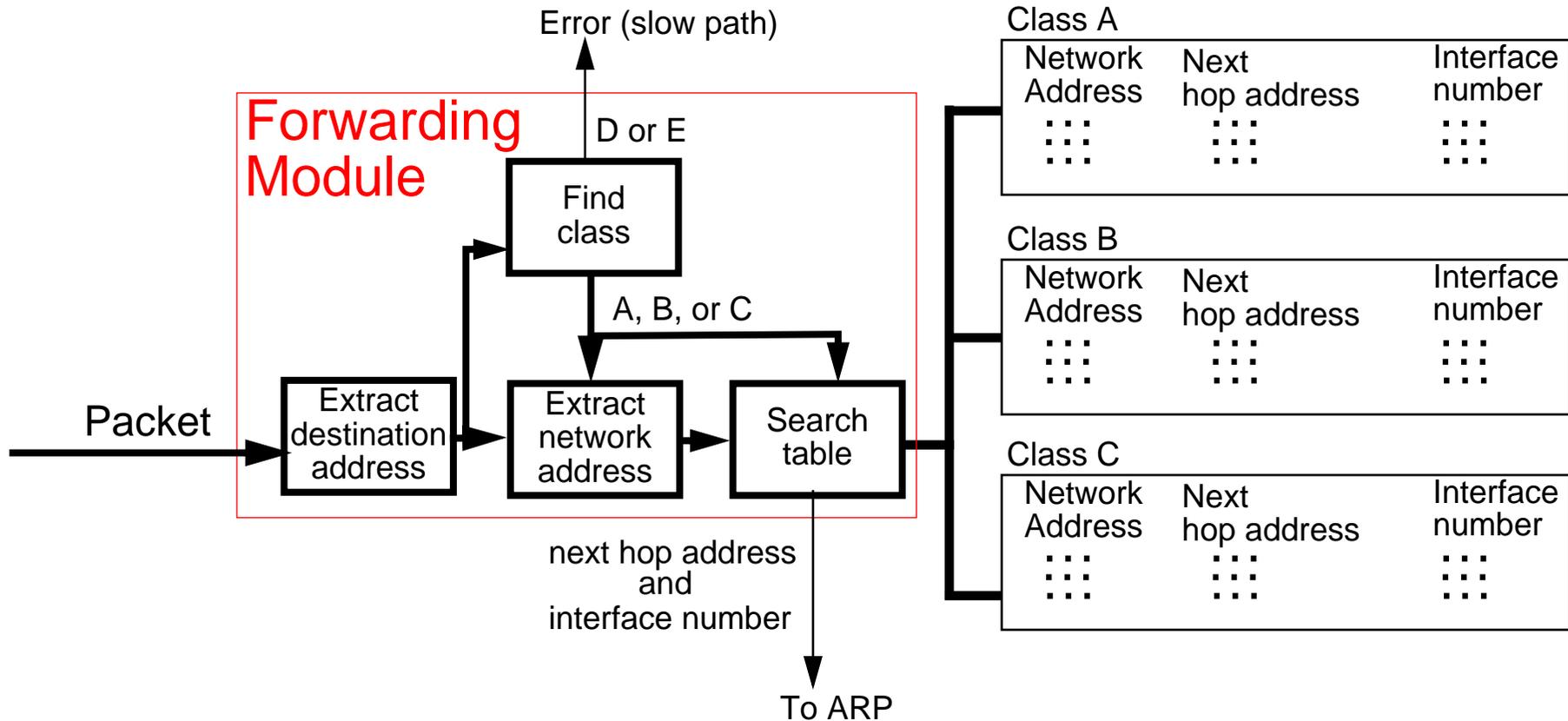
The bulk of the forwarding effort is **searching** the tables (as most of the rest of the processing is simple logical bit operations).

Maguire
maguire@it.kth.se

Forwarding module
2006.03.13

IP Basics: Routing, ARP, and RARP 82 of 121
Internetworking/Internetteknik

Routing Table Search - Classless

- ## Match destination with longest prefixes first
  - Software algorithms: tree, binary trees, tries (different data structures) [12]
  - Hardware support: Content Addressable Memory
    - Ternary CAM (TCAM) - [7],[8] - for searching and pattern matching
    - Ternary because each cell has three states: 0, 1, or ? (aka "don't care")
    - Performance an associate search (i.e., in parallel), returns the index (X) of the first match
    - 4-8 ns/search (i.e., 125M - 250M lookups per second)

0																																
1																																
2																																
…																																
X	1	0	0	0	0	1	0	1	1	1	0	0	0	0	0	1	0	1	1	1	0	0	0	?	?	?	?	?	?	?	?
…												Match																			
2^N-1																																
	1	0	0	0	0	1	0	1	1	1	0	0	0	0	0	1	0	1	1	1	0	0	0	0	0	1	0	1	1	1	1

index   address to search for (130.237.32.51)
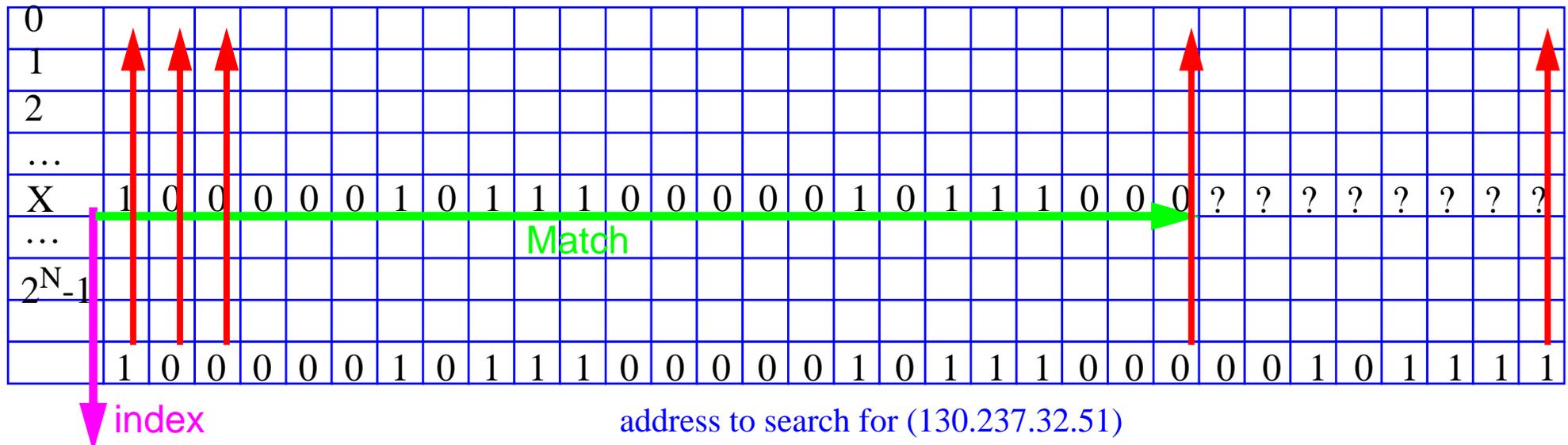
Note that the entries in the table have to be sorted - to achieve the longest match being at the smallest index.

Fast forwarding

Mikael Degermark, Andrej Brodnik, Svante Carlsson, Stephen Pink, "Small Forwarding Tables for Fast Routing Lookups", in Proceedings of the ACM SIGCOMM'97. (*compressed postscript*) {basis for *Effnet AB*}

- IP routing lookups must find routing entry with longest matching prefix.

Networking community *assumed* it was impossible to do IP routing lookups in software fast enough to support gigabit speeds - but they were wrong!

Paper presents a forwarding table data struct. designed for quick routing lookups.

- Such forwarding tables are small enough to fit in the cache of a conventional general purpose processor.
- The forwarding tables are very small, a large routing table with 40,000 routing entries can be compacted to a forwarding table of 150-160 Kbytes.

- With the table in cache, a 200 MHz Pentium Pro or 333 MHz Alpha 21164 can perform >2 million lookups per second.

- A lookup typically requires less than 100 instructions on an Alpha, using eight memory references accessing a total of 14 bytes.

∴ Full routing lookup of each IP packet at gigabit speeds without special hardware

Maguire
maguire@it.kth.se

Fast forwarding
2006.03.13

IP Basics: Routing, ARP, and RARP 84 of 121
Internetworking/Internetteknik

Routing Tables

- ## Aggregate IP addresses (i.e., exploit CIDR)
    - more specific networks (with longer prefixes)
    - less specific networks (with shorter prefixes)
    - $\Rightarrow$ smaller routing tables

- ## If each routing domain exports (i.e., tells others) only a small set of prefixes, this makes it easier for other routers to send traffic to it
    - Unfortunately this requires clever address assignments

- ## Some mechanisms lead to increased fragmentation
    - Due to limited availability of addresses long prefixes (particularily /24) are scattered geographically
    - Increasingly sites are connected to multiple ISPs (for redundancy) i.e., Multihoming
      - thus they have addresses from several different subnetworks

- ## Current routing tables have ~157,975 entries [13] (of which a large fraction are /24 prefixes) with a growth rate of "18,000 entries per year"[14].

There are a limited number of prefixes for Class A + B + C networks (2,113,664). If the longest prefixes which a backbone router had to deal with were /24, then a table with 16,777,216 entries would be sufficient (even without aggregation) - each entry only needs to store the outgoing port number! This would allow a direct lookup in a memory of ~26Mbytes - with upto 256 outgoing ports.

Maguire
maguire@it.kth.se

Routing Tables
2006.03.13

IP Basics: Routing, ARP, and RARP 85 of 121
Internetworking/Internetteknik

Routing table

Flags	Destination IP address	Next-hop Router IP address	point to local interface to use	Refcnt	Use	PMTU …
UGH	140.252.13.65	140.252.13.35	emd0	*ddd*	*ddd*	*ddd*
U	140.252.13.32	140.252.13.34	emd0	*ddd*	*ddd*	*ddd*
UG	default	140.252.13.33	emd0			
UH	127.0.0.1	127.0.0.1	lo0			

where *ddd* is some numeric value.

display the routing table with "netstat -rn"

"r" is for routing table

"n" asks for numeric IP addresses rather than name

Flags:

U	route is Up
G	route is to a Gateway
H	route is to a Host
D	route was Discovered by a redirect
M	route was Modified by a redirect

Maguire
maguire@it.kth.se

Routing table
2006.03.13

IP Basics: Routing, ARP, and RARP 86 of 121
Internetworking/Internetteknik

Host vs. router - two behaviors

- Hosts generate or sink IP packets
- Routers forward IP packets

Thus it is possible for a device to be both a host **and** a router.

Unless a host is **explicitly** configured as a router is **not** supposed to forward IP datagrams. The default behavior must be **never forward**.

In linux the variable which controls this is: `/proc/sys/net/ipv4/ip_forward`

- If this variable is set to **1**, then the node **will** perform IP forwarding.
- If this variable is set to **0**, then the node **will not** perform IP forwarding.

Maguire
maguire@it.kth.se

Host vs. router - two behaviors
2006.03.13

IP Basics: Routing, ARP, and RARP 87 of 121
Internetworking/Internetteknik

Host routing

A host either:

- knows a route - manually configured [i.e., "Static routes"]
  - from the interface (for directly connected networks) or manually via the "route" command
- or learns of a route [i.e., "Dynamic routes"]
  - Simplest method of learning a route:
    - The host sends a packet via the default route and is told via an ICMP Redirect of a better route
  - or the host hears an ICMP router advertisement (perhaps in response to its ICMP router solicitation message)
    - routers (almost) periodically broadcast or multicast advertisements of their existence and desire to provide routing service
    - format of ICMP router advertisement packet shown in Forouzan figure 9.18 on page 226
    - advertisements typically every 450..600 seconds
    - advertisements have a stated lifetime (typically 30 minutes)
  - or the host learns via a dynamic routing protocol.
- or uses a default route.

On booting hosts send ~3 ICMP router solicitation messages (~3 seconds apart) to find a default router. This allows for dynamic discovery of the default router.

Maguire
maguire@it.kth.se

Host routing
2006.03.13

IP Basics: Routing, ARP, and RARP 88 of 121
Internetworking/Internetteknik
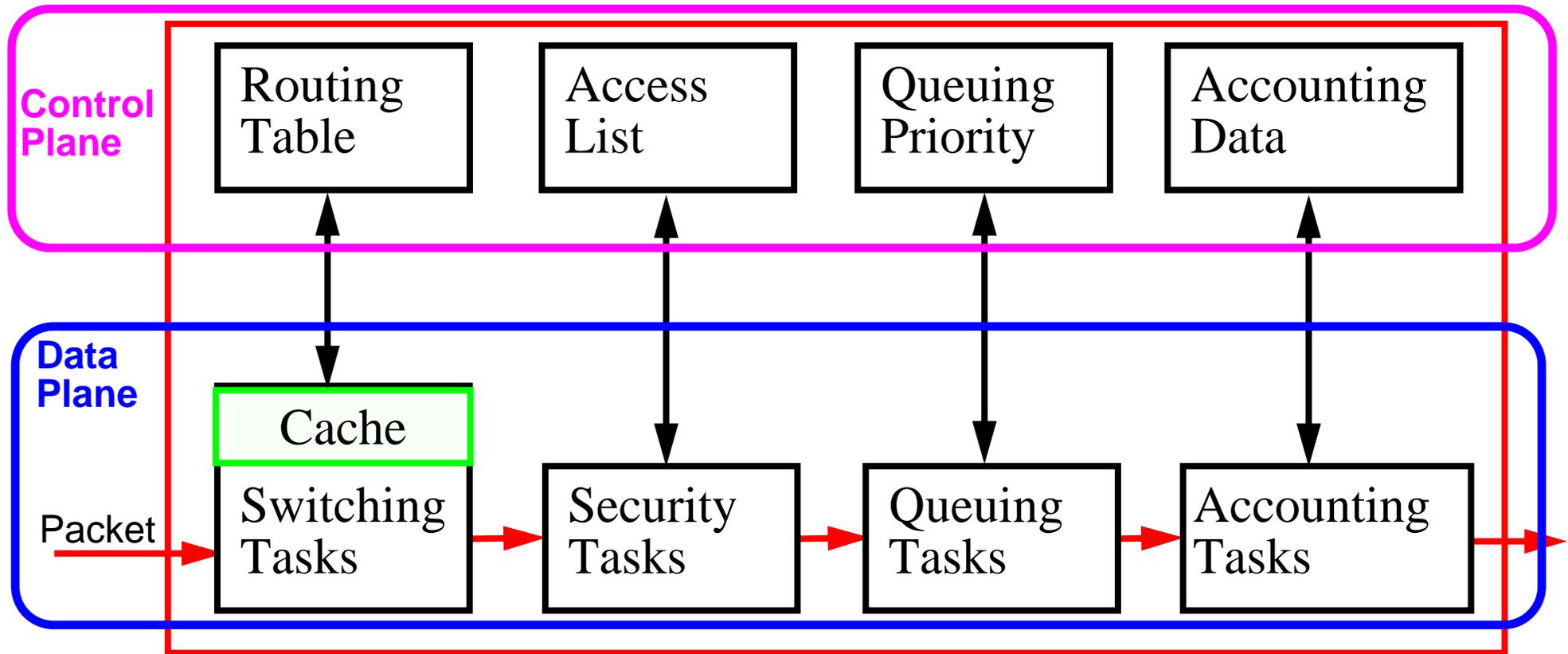
Routing

Figure 18: Basic steps in Routing

The routing table tells us which output port to use based on the destination (and possibly the source) IP address. The data plane has to run at packet rates (i.e., in real-time). However, a router also performs a lot of other processing

Maguire
maguire@it.kth.se

Routing
2006.03.13

IP Basics: Routing, ARP, and RARP 89 of 121
Internetworking/Internetteknik

Combining layers

Many devices now combine processing of several layers:

- Switch/Routers: combine layers 2+3

Devices combining layers 3+4 are appearing - which extract "flows" based on looking at transport layer port numbers in addition to network addresses.

Maguire
maguire@it.kth.se

Combining layers
2006.03.13

IP Basics: Routing, ARP, and RARP 90 of 121
Internetworking/Internetteknik

ARP and RARP

Address resolution (logical $\Leftrightarrow$ physical addresses):

- Mapping IP addresses $\Rightarrow$ link layer (MAC) addresses via Address Resolution Protocol (ARP)

- Mapping link layer (MAC) addresses $\Rightarrow$ IP addresses via Reverse Address Resolution Protocol (RARP)

Maguire
maguire@it.kth.se

ARP and RARP
2006.03.13

IP Basics: Routing, ARP, and RARP 91 of 121
Internetworking/Internetteknik

What to do with a new computer?

We will assume that the computer has an ethernet interface:

oscar.it.kth.se   ?hostname?   A new computer

130.237.212.253

08:00:20:7a:bc:2d

32 bit Internet address

48 bit Ethernet address
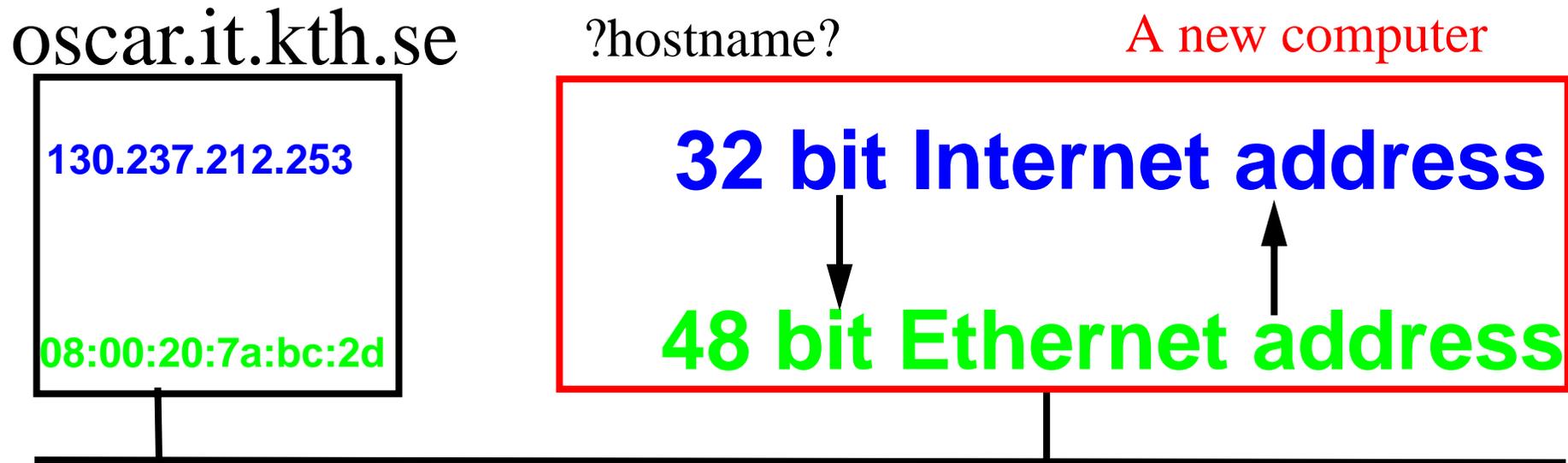
Figure 19: Name and IP Address needed!

- **Direct mapping** - requires no I/O, just a computation; hard to maintain; and requires stable storage (since you have to store the mappings somewhere) *or*

- **Dynamic Binding** - easier to maintain; but has a delay while messages are exchanged

Maguire
maguire@it.kth.se

What to do with a new computer? IP Basics: Routing, ARP, and RARP 92 of 121
2006.03.13
Internetworking/Internetteknik

Address Resolution: ARP, RARP

Host name: **string** "**ccslab1.kth.se**"

Name Resolution: DNS, Host File, …

IP address **32 bit** **130.237.15.254**

ARP

RARP

Address Resolution

HW address: **48 bit Ethernet MAC address**

"**08:00:2B:00:EE:0B**"
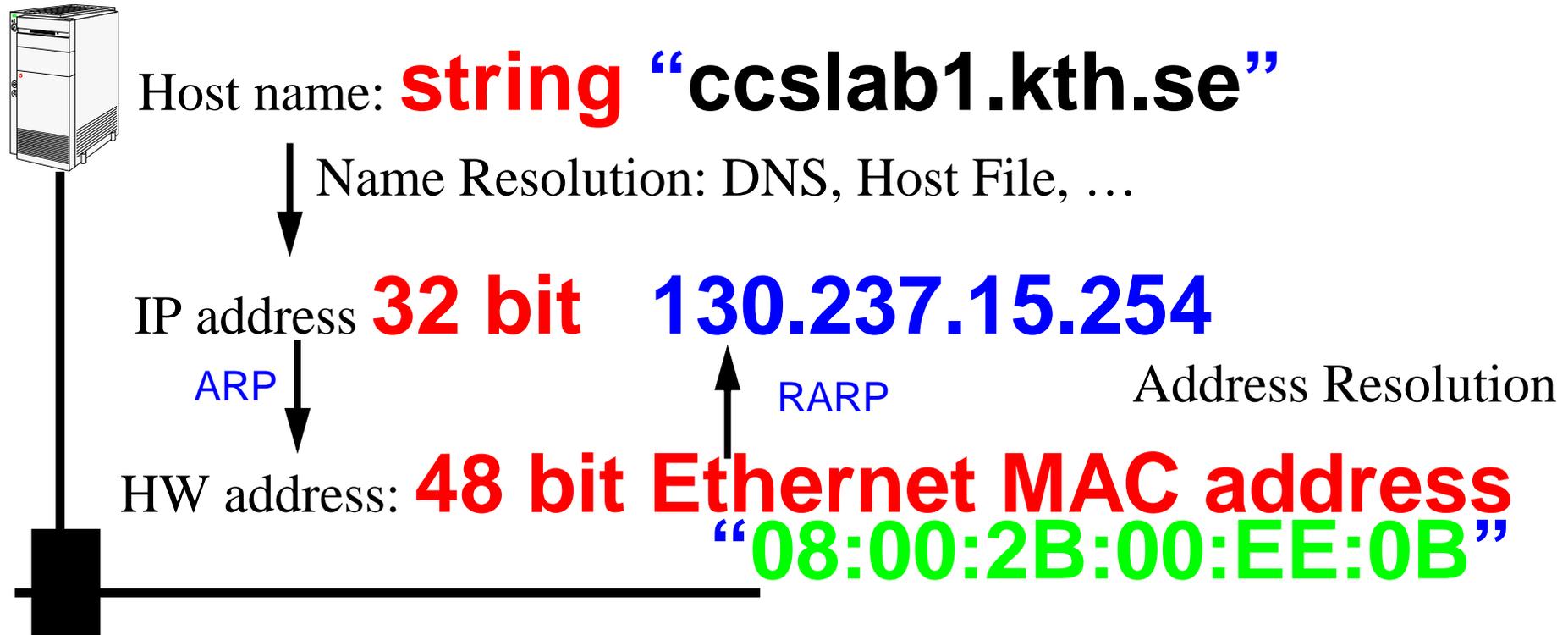
Figure 20: mapping between host names, IP address, and MAC address

ARP - Address Resolution Protocol

RARP- Reverse ARP

ARP ≡ Address Resolution Protocol (RFC826)

Address Resolution Protocol (ARP) - allows a host to find the physical address of a target host **on the same network**, given only the target's internet address.

- Sending host (source) wants to send an IP datagram, but does not know the corresponding ethernet address

- ARP request - broadcast to every host on the network (i.e., EtherDST=0xFFFFFFFFFFFF), TYPE=0x0806

- Destination host: "It is my address!" and sends an ARP reply

- Source host - receives the unicast ARP reply, and now uses it to send the IP datagram

EtherDST	EtherSRC	TYPE
6	6	2

OP=Request≡1, Reply≡2

hardware type	protocol type	hardware length = 6	protocol length = 4	OP	sender ether addr	sender IP	target ether addr	target IP
2	2	1	1	2	6	4	6	4
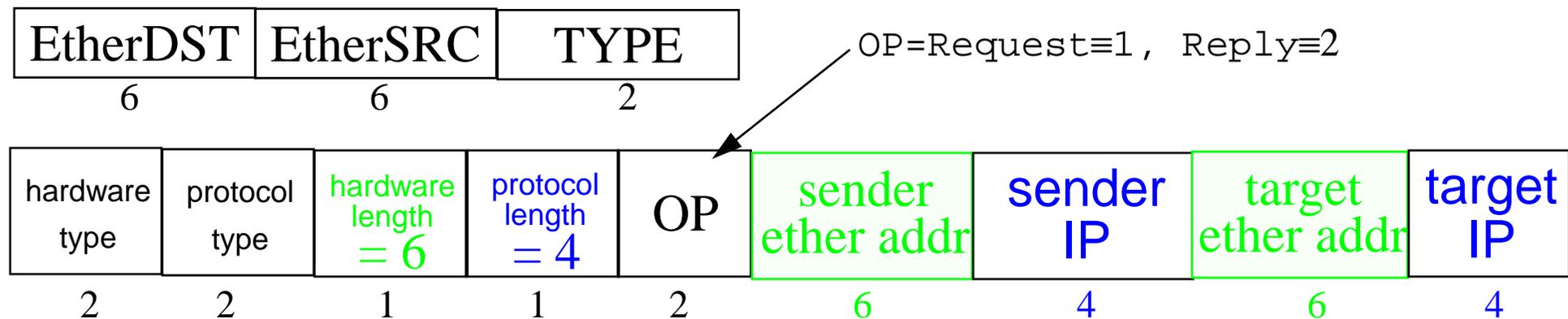
Figure 21: Format of ARP request/reply packet (see Stevens, Vol. 1, figure 4.3, pg. 56)

Maguire
maguire@it.kth.se

ARP ≡ Address Resolution Protocol (RFC826)IP Basics: Routing, ARP, and RARP 94 of
2006.03.13
Internetworking/Internetteknik

ARP example 1
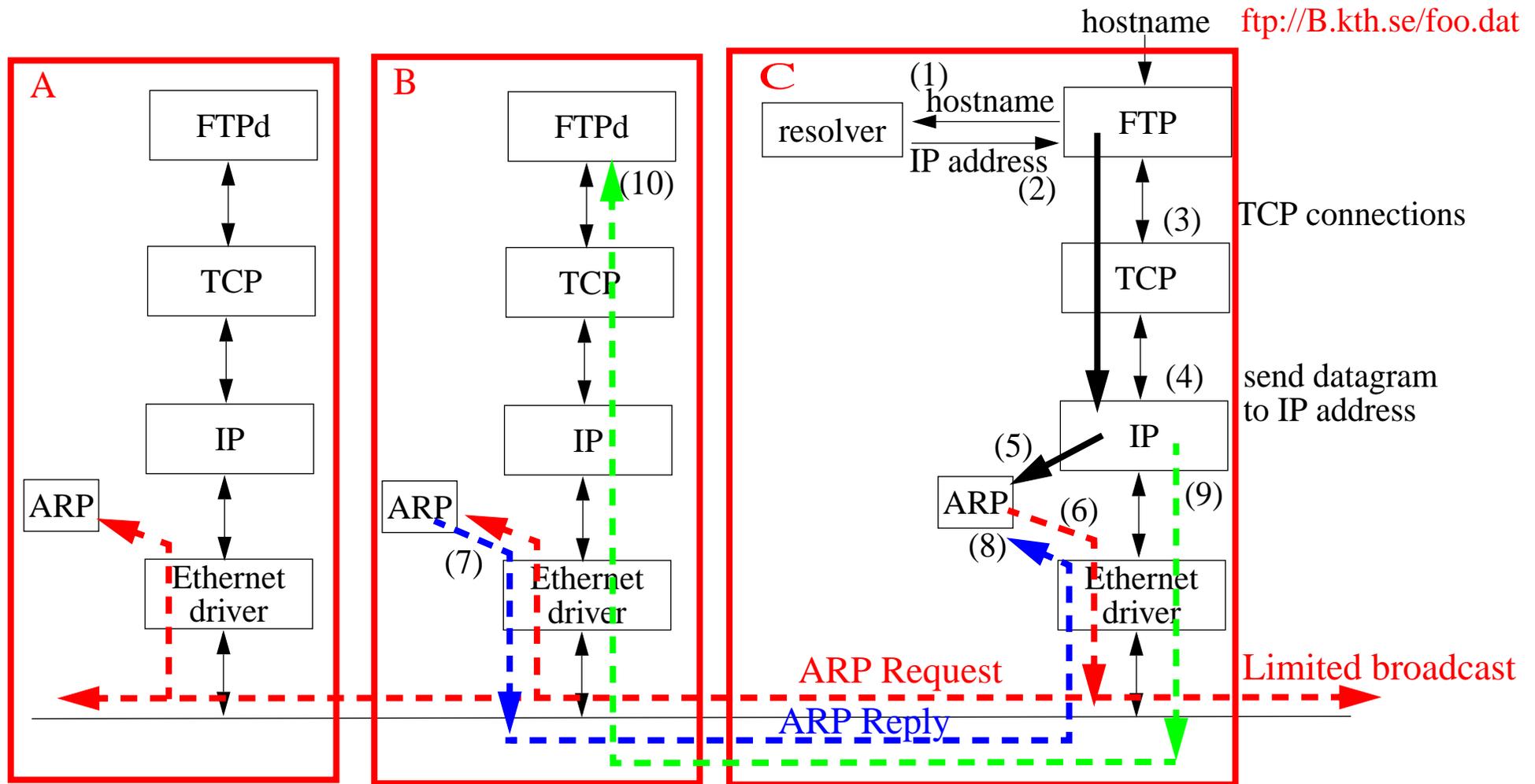
Figure 22: Using ARP on host C to determine MAC address of the interface on host B

Maguire
maguire@it.kth.se
ARP example 1
2006.03.13
IP Basics: Routing, ARP, and RARP 95 of 121
Internetworking/Internetteknik

Address Resolution Cache

Since you have just looked up the address, save (cache) it for reuse:

- to limit ARP traffic
- works because of correlations in use of addresses

You can examine the arp cache:

```
arp -a
machine-name (x.x.x.x) at xx:xx:xx:xx:xx:xx
…

arp -an
(x.x.x.x) at xx:xx:xx:xx:xx:xx
…
```

Note that the later form (with the "n" option) does *not* lookup the hostname, this is *very useful* when you don't yet have a name resolution service working!

ARP Refinements

Since the sender's Internet-to-Physical address binding is in every ARP broadcast; (all) receivers update their caches before processing an ARP packet

Maguire
maguire@it.kth.se
Address Resolution Cache
2006.03.13
IP Basics: Routing, ARP, and RARP 96 of 121
Internetworking/Internetteknik

ARP Timeouts

- ## If there is no reply to an ARP request
  - the machine is down or not responding
  - request was lost, then retry (but not too often)
  - eventually give up (When?)

- ## ARP cache timeouts
  - Berkeley implementations timeout
    - completed entry in 20 minutes
    - incomplete entry in 3 minutes
  - Linux:
    - for entries to which there has been no traffic a timeout occurs at `gc_stale_time`, set to 60 seconds by default (`/proc/sys/net/ipv4/neigh/default/gc_stale_time`)
  - Microsoft Windows NT
    - Using the registry editor, see HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters enter ArpCacheLife default value is set to 2 minutes
  - Cisco IOS v10.0 and above
    - select interface then "arp timeout xxxx", default value is set to 240 minutes (14400 sec.) and can be changed on a per-interface basis
  - RFC 1533: DHCP Options and BOOTP Vendor Extensions, defines:
    - ARP Cache Timeout Option (code for this option is 35).
  - Host Requirements RFC - says entries should be timed out **even if in use**

Maguire
maguire@it.kth.se

ARP Timeouts
2006.03.13

IP Basics: Routing, ARP, and RARP 97 of 121
Internetworking/Internetteknik

ARP example 2
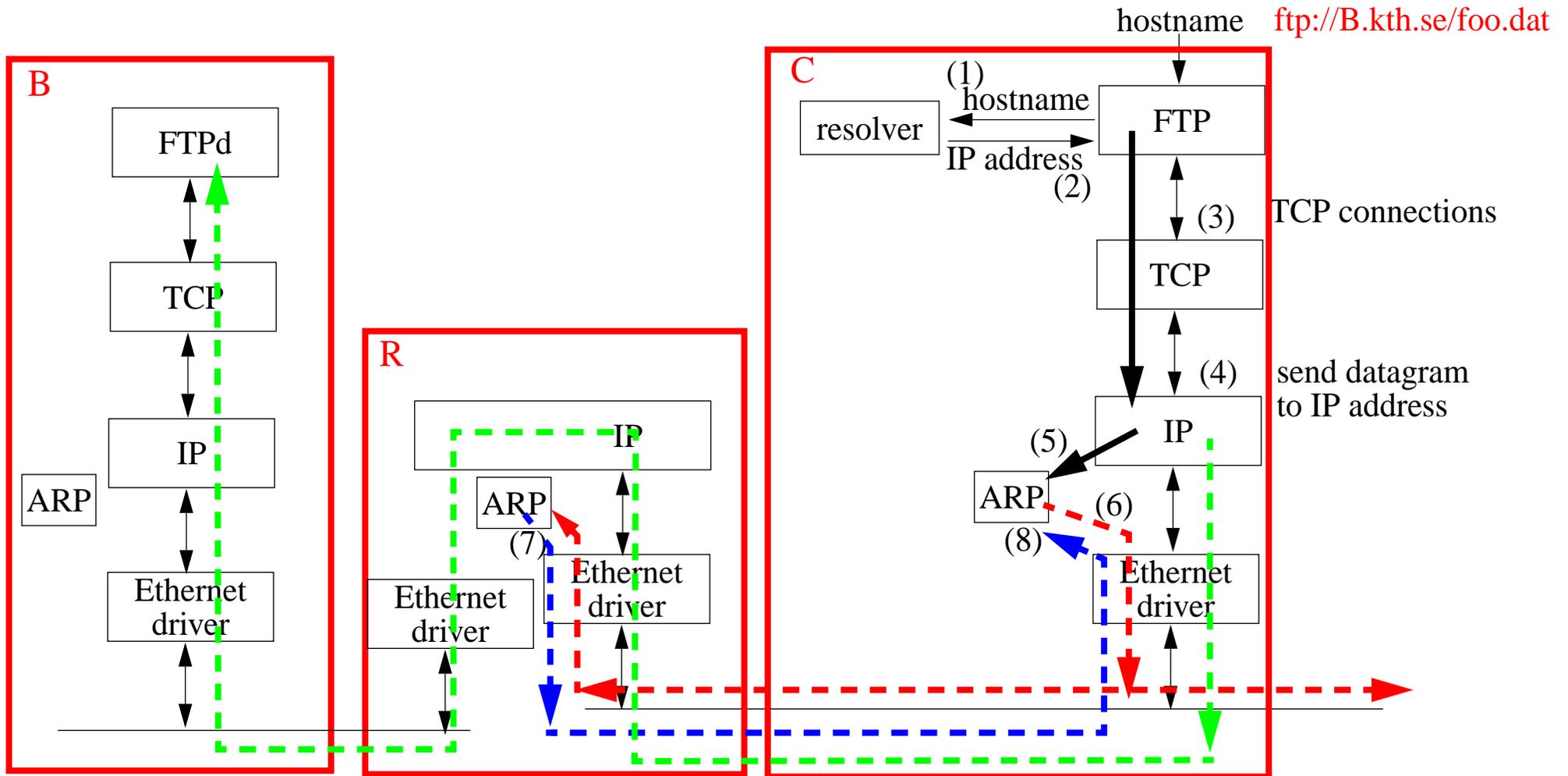
Figure 23: Router (R) doing a Proxy ARP to provide MAC address of B

Maguire
maguire@it.kth.se

ARP example 2
2006.03.13

IP Basics: Routing, ARP, and RARP 98 of 121
Internetworking/Internetteknik

Proxy ARP (RFC 826)

Lets a **router** on the network answer for a host which is NOT necessarily on the local network segment.

But how does this router know?

- It can make an ARP request itself or
- Perhaps it already knows - because it has an entry in it's ARP cache

For an example of using proxy arp with subnetting see:

- *http://www.linuxdoc.org/HOWTO/mini/Proxy-ARP-Subnet/why.html* and
- *http://www.linuxdoc.org/HOWTO/mini/Proxy-ARP-Subnet/how.html*

Maguire
maguire@it.kth.se

Proxy ARP (RFC 826)
2006.03.13

IP Basics: Routing, ARP, and RARP 99 of 121
Internetworking/Internetteknik

Gratuitous ARP

Host sends a request for its own address

- generally done at boot time to inform other machines of its address (possibly a new address) - gives these other hosts a chance to update their cache entries immediately

- lets hosts check to see if there is another machine claiming the same address $\Rightarrow$
  "duplicate IP address sent from Ethernet address a:b:c:d:e:f"

As noted before, hosts have paid the price by servicing the broadcast, so they can cache this information - this is one of the ways the proxy ARP server could know the mapping.

Note that faking that you are another machine can be used to provide **failover** for servers (see for example heartbeat, fake, etc. at *http://www.linux-ha.org/download/* for a send_arp program). [It can also be used very various attacks!]

Maguire
maguire@it.kth.se
Gratuitous ARP
2006.03.13
IP Basics: Routing, ARP, and RARP 100 of 121
Internetworking/Internetteknik

Additional ARP commands

- publish entries (i.e., mechanically make an entry and answer replies)

Publishing entires is one way that (embedded) devices can learn their IP addres.

```
# arp -s birkexample 08:00:2B:00:EE:0B pub
# arp -an
  (192.168.1.1) at 0:4:5a:de:e8:f9 ether
  …
  (172.16.32.20) at 8:0:2b:0:ee:b ether permanent published
```

where birkexample has the IP address: 172.16.32.20

- expliclity delete entries

```
# arp -d birkexample
birkexample (172.16.32.20) deleted
# arp -an
  (192.168.1.1) at 0:4:5a:de:e8:f9 ether
```

Maguire
maguire@it.kth.se

Additional ARP commands
2006.03.13

IP Basics: Routing, ARP, and RARP 101 of 121
Internetworking/Internetteknik

ARP - as seen with ethereal

Time	Source	Destination	Protocol	Info
1.995245	172.16.33.3	Broadcast	ARP	Who has 172.16.33.2? Tell 172.16.33.3

Frame 2 (60 bytes on wire, 60 bytes captured)
   Arrival Time: Mar 23, 2005 11:32:45.184792000
   Time delta from previous packet: 1.995245000 seconds
   Time since reference or first frame: 1.995245000 seconds

IEEE 802.3 Ethernet
   Destination: ff:ff:ff:ff:ff:ff (Broadcast)
   Source: 00:40:8c:30:d4:32 (172.16.33.3)
   Length: 36
   Trailer: 000000000000000000000
   Type: ARP (0x0806)
Address Resolution Protocol (request)
   Hardware type: IEEE 802 (0x0006)
   Protocol type: IP (0x0800)
   Hardware size: 6
   Protocol size: 4   Opcode: request (0x0001)
   Sender MAC address: 00:40:8c:30:d4:32 (172.16.33.3)
   Sender IP address: 172.16.33.3 (172.16.33.3)
   Target MAC address: ff:ff:ff:ff:ff:ff (Broadcast)
   Target IP address: 172.16.33.2 (172.16.33.2)

```
0000   ff ff ff ff ff ff 00 40 8c 30 d4 32 00 24 aa aa      .......@.0.2.$..
0010   03 00 00 00 08 06 00 06 08 00 06 04 00 01 00 40      ...............@
0020   8c 30 d4 32 ac 10 21 03 ff ff ff ff ff ff ac 10      .0.2..!..........   <<< unlike what page 163 says it is not all zeros!
0030   21 02 00 00 00 00 00 00 00 00 00 00                  !...........
```

non ARP example 1
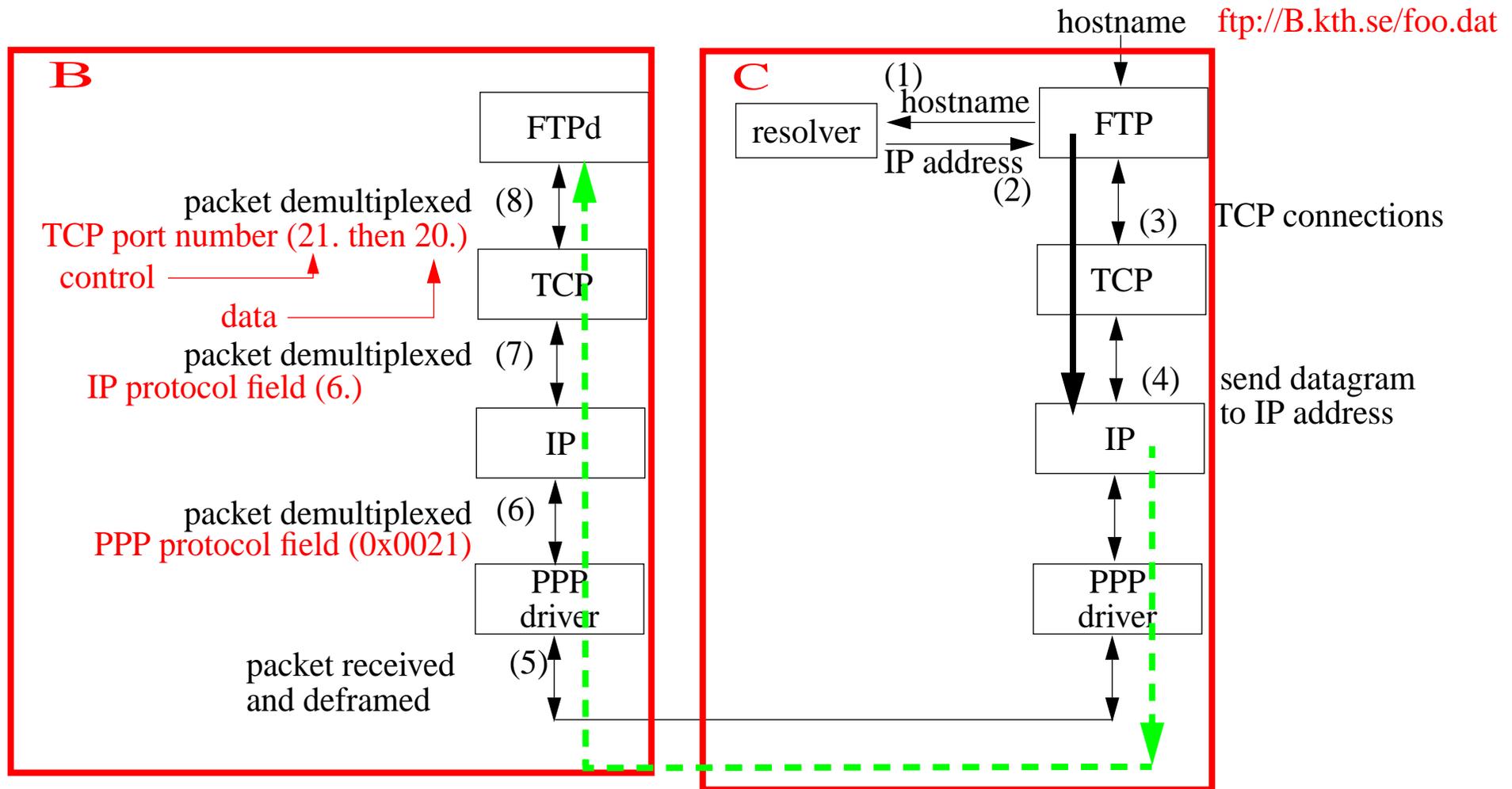
Figure 24: On a point-to-point link there is no need for ARP (figure also shows explicitly the demultiplexing)

Note that the PPP protocol field plays a role similar to the ethernet frame type.

Maguire
maguire@it.kth.se

non ARP example 1
2006.03.13

IP Basics: Routing, ARP, and RARP 103 of 121
Internetworking/Internetteknik

RARP: Reverse Address Resolution Protocol (RFC 903)

How do you get you own IP address, when all you know is your link address?

- Necessary if you don't have a disk or other stable store
- RARP request - broadcast to every host on the network (i.e., EtherDST=0xFFFFFFFFFFFF), TYPE=0x8035
- RARP server: "I know that address!" and sends an RARP reply
- Source host - receives the RARP reply, and now knows its own IP addr

EtherDST	EtherSRC	TYPE
6	6	2

OP=Request≡3, Reply≡4

hardware type	protocol type	hardware length	protocol length	OP	sender ether addr	sender IP	target ether addr	target IP
2	2	1	1	2	6	4	6	4
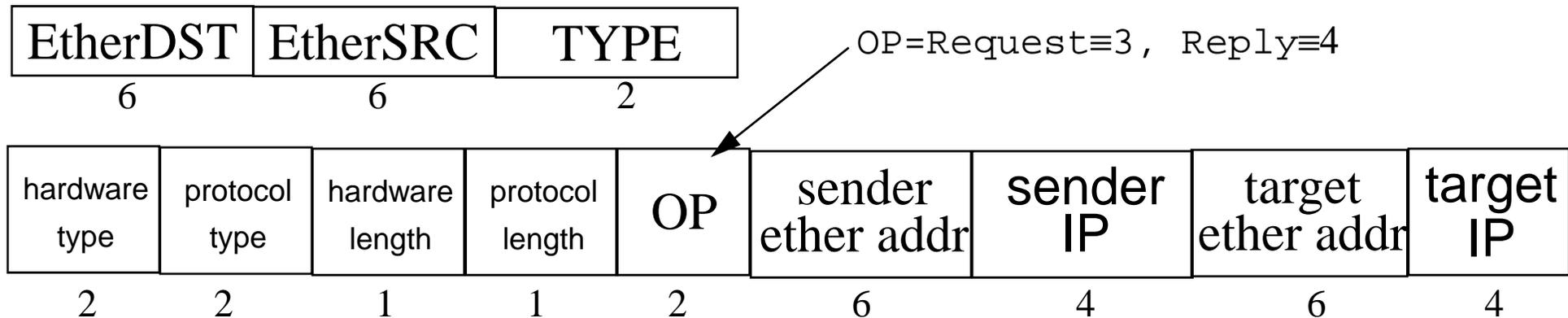
Figure 25: Format of RARP request/reply packet

Note: You can now see what the "publish" aspect of the `arp` command is for.

RARP - as seen with ethereal

Time	Source	Destination	Protocol	Info
0.000000	172.16.33.3	Broadcast	RARP	Who is 00:40:8c:30:d4:32? Tell 00:40:8c:30:d4:32

Frame 1 (60 bytes on wire, 60 bytes captured)
   Arrival Time: Mar 23, 2005 11:32:43.189547000
   Time delta from previous packet: 0.000000000 seconds
   Time since reference or first frame: 0.000000000 seconds

Ethernet II, Src: 00:40:8c:30:d4:32, Dst: ff:ff:ff:ff:ff:ff
   Destination: ff:ff:ff:ff:ff:ff (Broadcast)
   Source: 00:40:8c:30:d4:32 (172.16.33.3)
   Type: RARP (0x8035)
     Trailer: 00000000000000000000000000000000...

Address Resolution Protocol (reverse request)
   Hardware type: Ethernet (0x0001)
   Protocol type: IP (0x0800)
   Hardware size: 6
   Protocol size: 4
   Opcode: reverse request (0x0003)
   Sender MAC address: 00:40:8c:30:d4:32 (172.16.33.3)
   Sender IP address: 0.0.0.0 (0.0.0.0)
   Target MAC address: 00:40:8c:30:d4:32 (172.16.33.3)
   Target IP address: 0.0.0.0 (0.0.0.0)

```
0000  ff ff ff ff ff ff 00 40 8c 30 d4 32 80 35 00 01   .......@.0.2.5..
0010  08 00 06 04 00 03 00 40 8c 30 d4 32 00 00 00 00   .......@.0.2....   <<< as the source does not know its own IP address
0020  00 40 8c 30 d4 32 00 00 00 00 00 00 00 00 00 00   .@.0.2..........   <<< as the source does not know the target's IP address
0030  00 00 00 00 00 00 00 00 00 00 00 00               ............
```

RARP server

Someone has to know the mappings - quite often this is in a file "/etc/ethers"

Since this information is generally in a file, RARP servers are generally implemented as **user processes** (because a kernel process should **not** do file I/O!)

- Unlike ARP responses which are generally part of the TCP/IP implementation (often part of the kernel).

- How does the process get the packets - since they aren't IP and won't come across a socket?
  - BSD Packet filters
  - SVR4 Data Link Provider Interface (DLPI)
  - SUN's Network Interface Tap (NIT)
  - Interestingly in the appendix to RFC 903 an alternative to having data link level access was to have two IOCTLs, one that would "sleep until there is a translation to be done, then pass the request out to the user process"; the other means: "enter this translation into the kernel table"

- RARP requests are sent as hardware level broadcasts - therefore are not forwarded across routers:
  - multiple servers per segement - so in case one is down; the first response is used
  - having the router answer

Maguire
maguire@it.kth.se

RARP server
2006.03.13

IP Basics: Routing, ARP, and RARP 106 of 121
Internetworking/Internetteknik

Alternatives to RARP

In a later lecture we will examine:

- BOOTP and DHCP (for both IPv4 and IPv6) and
- autoconfiguration for IPv6.

Maguire
maguire@it.kth.se

Alternatives to RARP
2006.03.13

IP Basics: Routing, ARP, and RARP 107 of 121
Internetworking/Internetteknik

Novel IPX/SPX Addresses

Another approach to network addresses - which are tied to the MAC address

IPX/SPX == INternetwork Packet Exchange/Sequenced Packet Exchange

IPX address: 32 bits of network ID and 48 bits of host ID (the ethernet address)

Problems:

- There is no central authority for allocating the network IDs
    - ✗ So if you interconnect multiple IPX networks you may have to renumber every network
- If you change ethernet cards, you get a new address!
- Assumes that all machines are attached to a high capacity LAN.

Advantages

- You only have to assign network numbers, then the hosts figure out their own address. Simpler administration.

Novell NetWare provides: Service Advertising Protocol (SAP), Routing Information Protocol (RIP), and NetWare Core Protocol (NCP).

Maguire
maguire@it.kth.se
Novel IPX/SPX Addresses
2006.03.13
IP Basics: Routing, ARP, and RARP 108 of 121
Internetworking/Internetteknik

Useful tools

For looking at and generating packets!

Maguire
maguire@it.kth.se

Useful tools
2006.03.13

IP Basics: Routing, ARP, and RARP 109 of 121
Internetworking/Internetteknik

tcpdump

Under HP-UX 11.0

```
# ./tcpdump -i /dev/dlpi0
tcpdump: listening on /dev/dlpi0
```

22:25:43.217866 birk2.5900 > nucmed35.50251: . ack 3089200293 win 8080 (DF)

22:25:43.290636 birk2.5900 > nucmed35.50251: P 0:4(4) ack 1 win 8080 (DF)

22:25:43.360064 nucmed35.50251 > birk2.5900: . ack 4 win 32768

22:25:43.363786 birk2.5900 > nucmed35.50251: P 4:167(163) ack 1 win 8080 (DF)

22:25:43.364159 nucmed35.50251 > birk2.5900: P 1:11(10) ack 167 win 32768

22:25:43.543867 birk2.5900 > nucmed35.50251: . ack 11 win 8070 (DF)

22:25:43.577483 birk2.5900 > nucmed35.50251: P 167:171(4) ack 11 win 8070 (DF)

22:25:43.640052 nucmed35.50251 > birk2.5900: . ack 171 win 32768

22:25:43.643793 birk2.5900 > nucmed35.50251: P 171:334(163) ack 11 win 8070 (DF)

22:25:43.644132 nucmed35.50251 > birk2.5900: P 11:21(10) ack 334 win 32768

22:25:43.750062 birk2.5900 > nucmed35.50251: . ack 21 win 8060 (DF)

22:25:43.873349 birk2.5900 > nucmed35.50251: P 334:338(4) ack 21 win 8060 (DF)

22:25:43.940073 nucmed35.50251 > birk2.5900: . ack 338 win 32768

13 packets received by filter

0 packets dropped by kernel

Maguire
maguire@it.kth.se

tcpdump
2006.03.13

IP Basics: Routing, ARP, and RARP 110 of 121
Internetworking/Internetteknik

tcpdump - Linux

nucmed30:/home/maguire # /usr/sbin/tcpdump -i eth1

tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 96 bytes

14:21:52.736671 IP nucmed30.local.domain.must-p2p > jackb.ssh: P 1818006646:1818006726(80) ack 307068981 win 591>
14:21:52.737291 IP jackb.ssh > nucmed30.local.domain.must-p2p: P 1:113(112) ack 80 win 32768 <nop,nop,timestamp >
14:21:52.737917 IP nucmed30.local.domain.must-p2p > jackb.ssh: P 80:160(80) ack 113 win 5910 <nop,nop,timestamp >
14:21:52.802719 IP jackb.ssh > nucmed30.local.domain.must-p2p: . ack 160 win 32768 <nop,nop,timestamp 25983516 2>

…

14:21:57.782196 arp who-has jackscan tell nucmed30.local.domain
14:21:57.784218 arp reply jackscan is-at 00:40:8c:30:d4:32
14:21:57.784253 IP nucmed30.local.domain > jackscan: icmp 64: echo request seq 1
14:21:57.784971 IP jackscan > nucmed30.local.domain: icmp 64: echo reply seq 1
14:21:58.782187 IP nucmed30.local.domain > jackscan: icmp 64: echo request seq 2
14:21:58.782912 IP jackscan > nucmed30.local.domain: icmp 64: echo reply seq 2
14:21:59.783036 IP nucmed30.local.domain > jackscan: icmp 64: echo request seq 3
14:21:59.783759 IP jackscan > nucmed30.local.domain: icmp 64: echo reply seq 3

…

14:21:59.802600 IP jackb.ssh > nucmed30.local.domain.must-p2p: . ack 2864 win 32768 <nop,nop,timestamp 25984216 >
14:22:00.739485 IP nucmed30.local.domain.must-p2p > jackb.ssh: P 2864:2944(80) ack 897 win 5910 <nop,nop,timesta>

84 packets captured
84 packets received by filter
0 packets dropped by kernel

Maguire
maguire@it.kth.se
tcpdump - Linux
2006.03.13
IP Basics: Routing, ARP, and RARP 111 of 121
Internetworking/Internetteknik

Tools Used: tcpdump Program
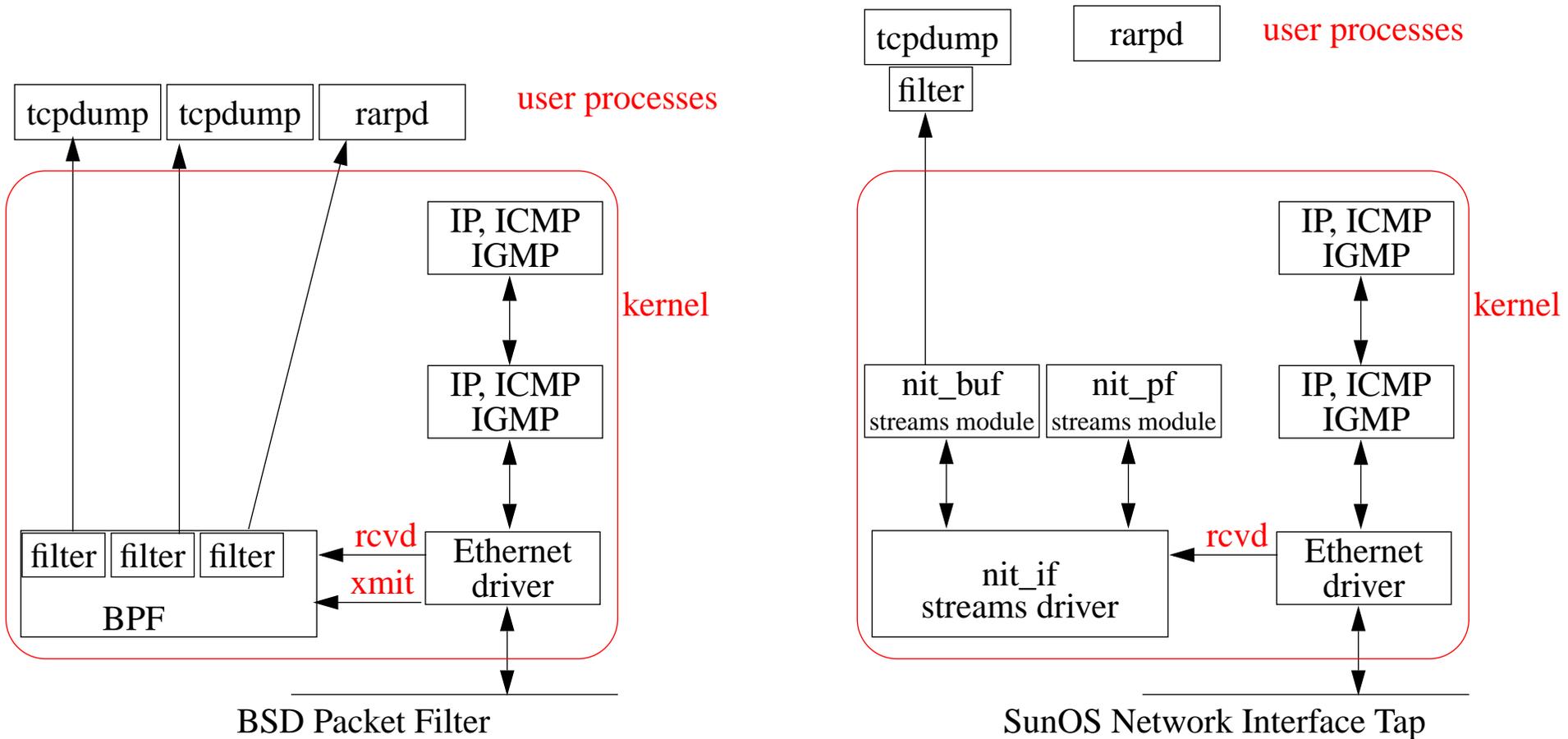
Figure 26: Two alternatives to get packets
Note the BSF packet filter gets a copy of both the received and transmitted packets.

Ethereal

Maguire
maguire@it.kth.se

Ethereal
2006.03.13

IP Basics: Routing, ARP, and RARP 113 of 121
Internetworking/Internetteknik

Linux Socket filter

If you want to sniff the network your self (with a program) - try the Linux Socket Filter [16] and [17].

Maguire
maguire@it.kth.se

Linux Socket filter
2006.03.13

IP Basics: Routing, ARP, and RARP 114 of 121
Internetworking/Internetteknik

Tools Used: sock Program

- A simple test program to generate TCP, UDP data
- To test and debug TCP, UDP implementations
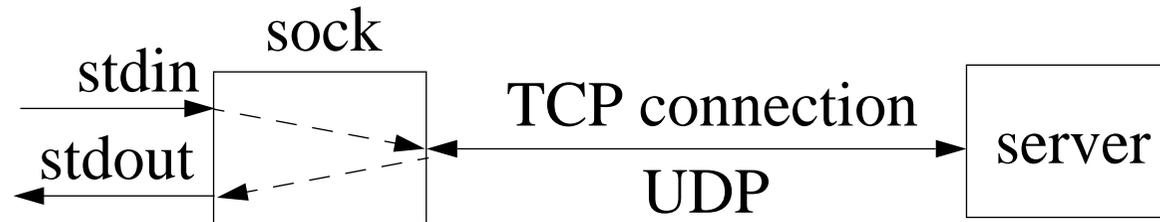
- Interactive client: default
- Interactive server: -s
- Source client: -i
- Sink server: -i -s
- Default TCP, -u for UDP

Source Code Available: (Tcpdump and sock)

For Win95/98/2000/NT: `http://netgroup-serv.polito.it/windump/`

For BSD alike: `ftp://ftp.uu.net/published/books/stevens.tcpipiv1.tar.Z`

Maguire
maguire@it.kth.se

Tools Used: sock Program
2006.03.13

IP Basics: Routing, ARP, and RARP 115 of 121
Internetworking/Internetteknik

Tools Used: sock + tcpdump
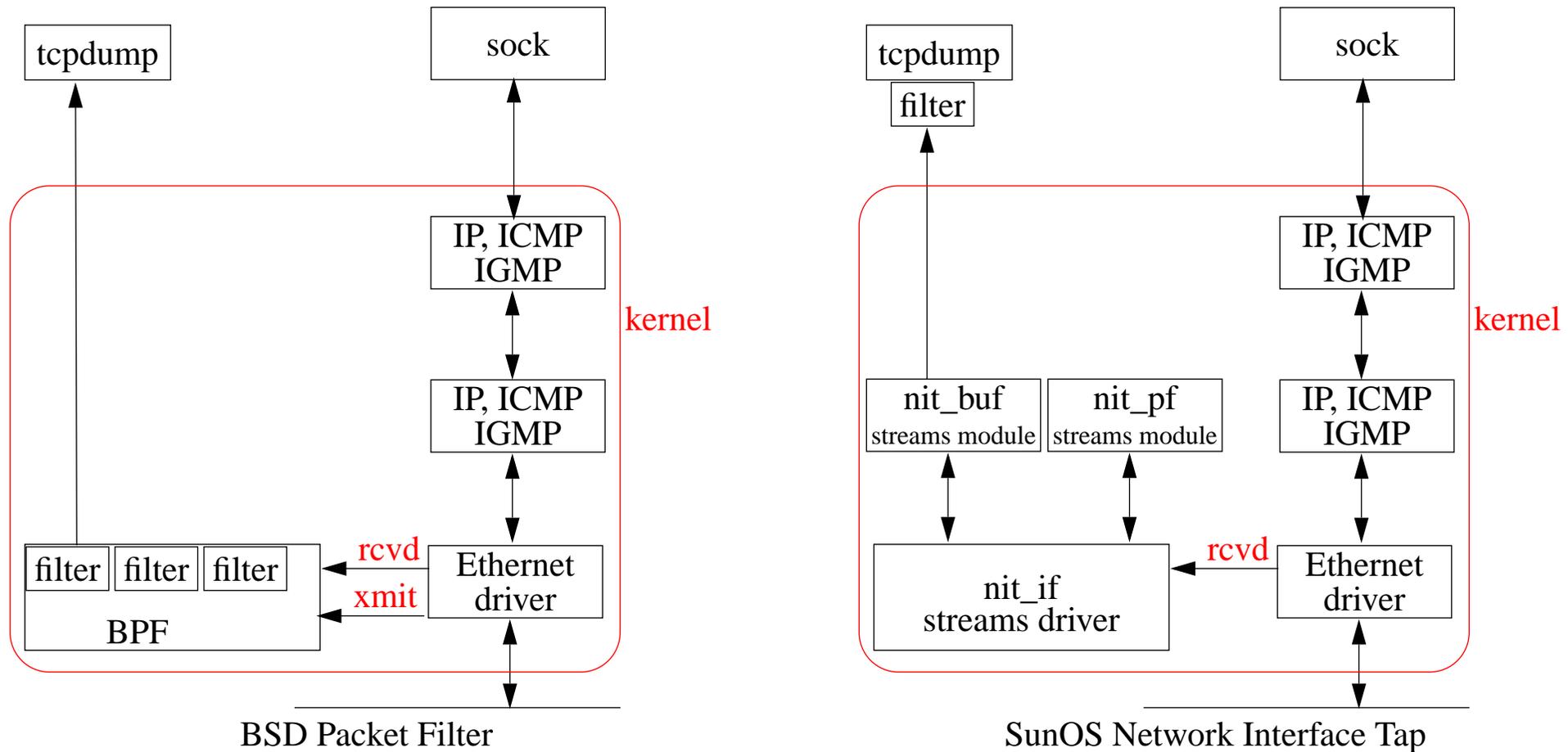
Figure 27: Two alternatives to generate and dump packets

Traffic generators

- Distributed Internet Traffic Generator (D-ITG) [18] -

  http://www.grid.unina.it/software/ITG/

- Gensyn generate multiple TCP streams in parallel

  http://www.item.ntnu.no/~poulh/GenSyn/gensyn.html

- Iperf *http://dast.nlanr.net/Projects/Iperf/*

- MGEN: network performance tests and measurements using UDP/IP traffic *http://mgen.pf.itd.nrl.navy.mil/*

- RUDE & CRUDE - Real-time UDP Data Emitter (RUDE) and Collector for RUDE (CRUDE) *http://rude.sourceforge.net/*

- SUN's Packet Shell - *http://playground.sun.com/psh/*

- TG *http://www.caip.rutgers.edu/~arni/linux/tg1.html*

- UDPgen *http://www.fokus.fhg.de/usr/sebastian.zander/private/udpgen*

- Netcom's SmartBits - hardware tester

For additional traffic generators see: *http://www.icir.org/models/trafficgenerators.html* and

http://www.ip-measurement.org/

Maguire
maguire@it.kth.se

Traffic generators
2006.03.13

IP Basics: Routing, ARP, and RARP 117 of 121
Internetworking/Internetteknik

Summary

This lecture we have discussed:

- ## Routing Principles
  - Routing Mechanism: Use the *most specific* route
    - IP provides the mechanism to route packets
  - Routing Policy: What routes should be put in the routing table?
    - Use a routing daemon to provide the *routing policy*

- ## Routing table

- ## ARP and RARP

- ## IPX/SPX Addresses - we will see something similar when we talk about IPv6

- ## tcpdump, ethereal, sock

For further information about routing see:

   Bassam Halabi, *Internet Routing Architectures*, Cisco Press, 1997, ISBN 1-56205-652-2. -- especially useful for IGRP.

We will examine routing policies and algorithms in a later lecture.

Maguire
maguire@it.kth.se

Summary
2006.03.13

IP Basics: Routing, ARP, and RARP 118 of 121
Internetworking/Internetteknik

References

[7]   Renesas Technology Corp. TCAM description

    http://www.renesas.com/fmwk.jsp?cnt=tcam_series_landing.jsp&fp=/applications/network/network_memory/tcam/

[8]   Fany Yu, Randy H. Katz, and T. V. Lakshman, "Gigabit Rate Multiple-Pattern Matching with TCAM",

    http://sahara.cs.berkeley.edu/jan2004-retreat/slides/Fang_retreat.ppt

[9]   Geoff Huston, "Analyzing the Internet BGP Routing Table", Cisco Systems web page,

    http://www.cisco.com/en/US/about/ac123/ac147/ac174/ac176/about_cisco_ipj_archive_article09186a00800c83cc.html

[10]   Tian Bu, Lixin Gao, and Don Towsley, "On Characterizing BGP Routing Table Growth", Proceedings of Globe Internet 2002, 2002

    http://www-unix.ecs.umass.edu/~lgao/globalinternet2002_tian.pdf

[11]   H. Narayan, R. Govindan, and G. Varghese, "The Impact of Address

Maguire
maguire@it.kth.se

References
2006.03.13

IP Basics: Routing, ARP, and RARP 119 of 121
Internetworking/Internetteknik

Allocation and Routing on the Structure and Implementation of Routing Tables", Proceedings of the 2003 Conference on Applications, technologies, architectures, and protocols for computer communications, 2003, pp 125-136, ISBN:1-58113-735-4 and SIGCOMM 03, August 25 29, 2003, Karlsruhe, Germany *http://www.cs.ucsd.edu/~varghese/PAPERS/aram.pdf*

[12] Ravikumar V.C Rabi Mahapatra J.C. Liu, "Modified LC-Trie Based Efficient Routing Lookup",

http://faculty.cs.tamu.edu/rabi/Publications/Mascot-final-proceeding.pdf

[13] APNIC, Routing Table Report 04:00 +10GMT Sat 19 Mar, 2005, North American Network Operators Group, Weekly Routing Table Report, From: Routing Table Analysis, Mar 18 13:10:37 2005, "This is an automated weekly mailing describing the state of the Internet Routing Table as seen from APNIC's router in Japan. Daily listings are sent to bgp-stats@lists.apnic.net" *http://www.merit.edu/mail.archives/nanog/2005-03/msg00401.html*

[14] Geoff Huston, Routing Table Status Report, Policy SIG, APNIC19, Kyoto, Japan, Feb 24 2005

Maguire
maguire@it.kth.se

References
2006.03.13

IP Basics: Routing, ARP, and RARP 120 of 121
Internetworking/Internetteknik

http://www.apnic.net/meetings/19/docs/sigs/routing/routing-pres-info-huston-routing-table.pdf

[15] Bassam Halabi, *Internet Routing Architectures*, Cisco Press, 1997, ISBN 1-56205-652-2.

[16] Gianluca Insolvibile, "The Linux Socket Filter: Sniffing Bytes over the Network", Linux Journal, 31 May 2001 *http://www.linuxjournal.com/article/4659*

[17] Gianluca Insolvibile, "Inside the Linux Packet Filter, Part II", Linux Journal, 1 March 2002 *http://www.linuxjournal.com/article/5617*

[18] Stefano Avallone, Antonio Pescapé, and Giorgio Ventre, "Analysis and experimentation of Internet Traffic Generator", International Conference on Next Generation Teletraffic and Wired/Wireless Advanced Networking (NEW2AN'04), February 02-06, 2004

http://www.grid.unina.it/software/ITG/D-ITGpubblications/New2an-ITG.pdf

Maguire
maguire@it.kth.se

References
2006.03.13

IP Basics: Routing, ARP, and RARP 121 of 121
Internetworking/Internetteknik

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 3:   IP, ICMP, and Tools

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapters 8-9

KTH Information and Communication Technology

IP, ICMP, and Tools Outline

- IP
- ICMP
- Useful Diagnostic Tools
  - Ping
  - Traceroute
  - tcpdump and ethereal
  - sock

Maguire
maguire@it.kth.se

IP, ICMP, and Tools Outline
2006.03.13

IP, ICMP, and Tools 123 of 152
Internetworking/Internetteknik

Internet Protocol version 4 (IPv4) (RFC 791)

0	7	8	15	16	23	24	31

4 bit version	4-bit header length	8-bit DS or Type of Service (TOS)	16 bit total length
16 bit identification		3-bit flags	13 bit Fragment Offset
8-bit Time to Live (TTL)	8-bit Protocol	16 bit header checksum	
32 bit Source IP address			
32 bit Destination IP address			
options (padded to 32 bit length)			
data			

20 bytes

Figure 28: IP header (see Stevens, Vol. 1, figure 3.1, pg. 34)

Note: We examined the Version, Protocol, and IP address fields in the previous lecture. Today we will examine the length fields, TOS, identification, flags, offset, checksum, and options fields.

Maguire
maguire@it.kth.se

Internet Protocol version 4 (IPv4) (RFC 791)
2006.03.13

IP, ICMP, and Tools 124 of 152
Internetworking/Internetteknik

Length Fields

- ## Header Length (4 bits)
  - Size of IPv4 header including IP options
  - Expressed in number of 32-bit words (4-byte words)
  - Minimum is 5 words (i.e., 20 bytes)
  - Maximum is 15 words (i.e., 60 bytes)
    - limited size $\Rightarrow$ limited use

- ## Total Length (16 bits)
  - Total length of datagram **including** header
  - If datagram is fragmented: length of this fragment
  - Expressed in bytes
  - Hosts **only** have to accept packets up to 576 bytes in size
  - Maximum: 65,535 bytes
    - Most modern systems accept slightly larger than 8,196 + header bytes (to provide efficient file service for 8 Kbyte blocks)
    - Note: Some systems **only** accept this much!

Maguire
maguire@it.kth.se

Length Fields
2006.03.13

IP, ICMP, and Tools 125 of 152
Internetworking/Internetteknik

MTU≡Maximum Transmission Unit

MTU is a characteristic of the link layer

Typical MTUS

MTU	Network	
65535	Official maximum MTU	
17914	16Mbps IBM Token Ring	
8166	IEEE 802.4	
4464	IEEE 802.5 (4Mbps max)	
4352	FDDI (Revised)	
2048	Wideband Network	
2002	IEEE 802.5 (4Mb recommended)	
1536	Experimental Ethernet Nets	
1500	Ethernet Networks	
1500	Point-to-Point (default)	
1492	IEEE 802.3	
1006	SLIP	simply a logical limit for interactive response
1006	ARPANET	
576	X.25 Networks	⇐ we will see this number again!
544	DEC IP Portal	
512	NETBIOS	
508	IEEE 802/Source-Route Bridge	
296	Point-to-Point (low delay)	
68	Official minimum MTU	

Maguire
maguire@it.kth.se

MTU≡Maximum Transmission Unit
2006.03.13

IP, ICMP, and Tools 126 of 152
Internetworking/Internetteknik

Fragmentation

- If an IP datagram is larger than the MTU of the link layer, it must be divided into several pieces $\Rightarrow$ fragmentation

- Fragmentation may occur multiple times
  - as a fragment might need to go across a link with an even smaller MTU!

- Both hosts and routers may fragment
  - **However, only** destination host reassemble!
    - as fragments only need to come together at the final host - thus the fragments can take different paths though the network
    - also reassembly requires waiting for the other fragments - so doing this earlier in the net-work could add unnecessary delay
  - Each fragment is routed separately (i.e., as independent datagram)

- TCP uses either 576 byte MTU or path MTU discovery

Maguire
maguire@it.kth.se

Fragmentation
2006.03.13

IP, ICMP, and Tools 127 of 152
Internetworking/Internetteknik

Fields relevant to Fragmentation

- ## Identification (16 bits)
  - Identification + source IP address *uniquely* identifies each datagram sent by a host ⇒ Identification field is copied to all fragments of a datagram upon fragmentation (since they are all part of the same original datagram)

- ## Flags: 3 bits
  - Reserved Fragment (RF) - set to 0
  - Don't Fragment (DF)
    - Set to 1 if datagram should **not** be fragmented
    - If set and fragmentation needed ⇒ datagram will be **discarded** and an **error message** will be returned to the sender
  - More Fragments (MF)
    - Set to 1 for all fragments, **except** the last

- ## Fragmentation Offset (13 bits)
  - 8-byte units: (i.e., the byte offset is ip_frag << 3)
  - indicates relative position of a fragment with respect to the whole datagram

Fragments can overlap - the receiver simply assembles what it receives (ignoring duplicate parts).

If there are gaps - then at some point there will be a re-assembly error.

Maguire
maguire@it.kth.se

Fields relevant to Fragmentation
2006.03.13

IP, ICMP, and Tools 128 of 152
Internetworking/Internetteknik

Path MTU

- Each link in path from source to destination can have a different MTU
- to avoid fragmentation you have to find the minimum of these
- RFC 1191: Path MTU discovery[20] uses:
    - "good" guesses (i.e., likely values)
    - By setting Don't Fragment (DF) bit in IP datagram $\Rightarrow$ change size while you get ICMP messages saying "Destination Unreachable" with a code saying fragmentation needed

Maguire
maguire@it.kth.se

Path MTU
2006.03.13

IP, ICMP, and Tools 129 of 152
Internetworking/Internetteknik

Serial line throughput

At 9,000 bits/sec, 8 bits per byte, plus 1 start and 1 stop bit, i.e., 960 bytes/sec, then transferring 1024 byte packets would take 1066 ms

- too long for interactive limits; since the average wait would be 533 ms

$\therefore$ shorten the MTU to 296 bytes $\Rightarrow$ 266 ms/frame or ~133 ms average wait

With 5 bytes of CSLIP header and 256 bytes of data (in the 261 byte frame) $\Rightarrow$

- 98.1% utilization of link for data and
- 1.9% for header

For single bytes of interactive traffic, the round trip-time is 12.5 ms

Caveats:

- assumes that you give interactive traffic priority
- error correcting and compression in the modem can complicate the calculations - since the modem has to delay traffic to have more to compress and compression takes time

Maguire
maguire@it.kth.se

Serial line throughput
2006.03.13

IP, ICMP, and Tools 130 of 152
Internetworking/Internetteknik

Differentiated Services (DS) & Type of Service

Type of Service (TOS): 8 bits

Bits 0-2: Precedence

Bit 3: 0 = normal Delay 1 = Low Delay

Bit 4: 0 = normal Throughput 1 = High Throughput

Bit 5: 0 = normal Relibility 1 = High Relibility

Bit 6: 0 = normal monetary Cost 1 = minimize monetary Cost.

0	1	2	3	4	5	6	7
Precedence			DELAY	T	R	C	Reserved

- Few applications set the TOS field (in fact most implementations will not let you set these bits!) However, 4.3BSD Reno and later - do support these bits.
- Differentiated Services (diffserv) proposes to use 6 of these bits to provide 64 priority levels - calling it the Differentiated Service (DS) field [RFC2474] (using bits 0..5 as Differentiated Services CodePoint (DSCP))
- SLIP guesses by looking at the **protocol** field and then checks the source and destination **port** numbers.

There has been a lot of experimentation with this field, both for TOS and more recently for Early Congestion Notification (ECN): RFC 3168 [19] using bits 6 and 7 {ECN Capable Transport (ECT) and Congestion Experienced (CE)}.

Maguire
maguire@it.kth.se

Differentiated Services (DS) & Type of Service
2006.03.13

IP, ICMP, and Tools 131 of 152
Internetworking/Internetteknik

Recommended Value for TOS Field

Application		Minimum delay	Maximize throughput	Maximize reliability	Minimize monitary cost	Hex value[a]
telnet/rlogin		1	0	0	0	0x10
FTP	control	1	0	0	0	0x10
	data	0	1	0	0	0x08
	any bulk data	0	1	0	0	0x08
TFTP		1	0	0	0	0x10
SMTP	command phase	1	0	0	0	0x10
	data phase	0	1	0	0	0x08
DNS	UDP query	1	0	0	0	0x10
	TCP query	0	0	0	0	0x00
	zone transfer	0	1	0	0	0x08
ICMP	error	0	0	0	0	0x00
	query	0	0	0	0	0x00
any IGP		0	0	1	0	0x04
SNMP		0	0	1	0	0x04
BOOTP		0	0	0	0	0x00
NNTP		0	0	0	1	0x02

a. Note that this is the hex value as see in the TOS/DS byte.

See also Table 8.2 on page 182 of Forouzan.

Maguire
maguire@it.kth.se

Recommended Value for TOS Field
2006.03.13

IP, ICMP, and Tools 132 of 152
Internetworking/Internetteknik

Precedence

Precedence values are defined but are largely ignored, few applications use them.

111	Network Control
110	Internetwork Control
101	CRITIC/ECP
100	Flash Override
011	Flash
010	Immediate
001	Priority
000	Routine

In the original ARPANET there were two priority levels defined (in order to support low delay services and regular traffic).

Maguire
maguire@it.kth.se

Precedence
2006.03.13

IP, ICMP, and Tools 133 of 152
Internetworking/Internetteknik

Problems with precedence

- As soon as people found that high priority meant something
  $\Rightarrow$ all traffic was sent with this bit set!

So unless there is a added cost/policy check/… associated with usage of a precedence level - it is very likely going to be abused.

Maguire
maguire@it.kth.se
Problems with precedence
2006.03.13
IP, ICMP, and Tools 134 of 152
Internetworking/Internetteknik

Precendence and telephony systems

Similar precedence systems exist in most national telephony systems.

Q: What are the A, B, C and D touch tone keys used for? …

A: These are extensions to the standard touch-tones (0-9, *, #) which originated with the U.S. military's Autovon phone network. The original names of these keys were FO (Flash Override), F (Flash), I (Immediate), and P (Priority). The various priority levels established calls with varying degrees of immediacy, terminating other conversations on the network if necessary. FO was the greatest priority, normally reserved for the President or very high ranking officials. P had a lesser priority, but still took precedence over calls that were placed without any priority established.

-- from TELECOM Digest - Frequently Asked Questions - v.8, 8 February 1997

Maguire
maguire@it.kth.se

Precendence and telephony systems
2006.03.13

IP, ICMP, and Tools 135 of 152
Internetworking/Internetteknik

Differentiated services

If bits 3, 4, and 5 are all zero (i.e., XXX000) $\Rightarrow$ treat the bits 1, 2, 3 as the traditional precedence bits, else the 6 bits define 64 services:

- Category 1: numbers 0, 2, 4, … 62 - defined by IETF
- Category 2: numbers 3, 7, 11, 15, … 63 defined by local authorities
- Category 3: numbers 1, 5, 9, … 61 are for temporary/experimental use

The numbering makes more sense when you see them as bit patterns:

Category	codepoint	Assigning authority
1	XXXXX0	IETF
2	XXXX11	local
3	XXXX01	temporary/experimental

The big problems occur at gateways where the intepretation of local DS values is different on the incoming and outgoing links!

Maguire
maguire@it.kth.se

Differentiated services
2006.03.13

IP, ICMP, and Tools 136 of 152
Internetworking/Internetteknik

TTL field

Time To Live (TTL) (8 bits):

- Limits the lifetime of a datagram, to avoid infinite loops
- A router receiving a packet with TTL>1 decrements the TTL field and forwards the packet
- If TTL <= 1 shall not be forwarded
  ⇒ an ICMP time exceeded error is returned to the sender {we will cover ICMP shortly}
- Recommended value is 64
- Should really be called Hop Limit (as in IPv6)

Historically: Every router holding a datagram for more than 1 **second** was expected to decrement the TTL by the *number of seconds* the datagram resided in the router.

Maguire
maguire@it.kth.se

TTL field
2006.03.13

IP, ICMP, and Tools 137 of 152
Internetworking/Internetteknik

Header Checksum

- ## Ensures integrity of header fields
  - Hop-by-hop (not end-to-end)
  - Header fields must be correct for proper and safe processing of IP!
  - Payload is **not** covered

- ## Other checksums
  - Hop-by-hop: using link-layer CRC
    - IP assumes a strong link layer checksum/CRC - as the IP checksum is weak
  - End-to-end: Transport layer checksums, e.g., TCP & UDP checksums, cover **payload**

- ## Internet Checksum Algorithm, RFC 1071
  - Treat headers as sequence of 16-bit integers
  - Add them together
  - Take the one's complement of the result

Note that recent work concerning IP over wireless links assume that the payload can have errors and will still be received (see work concerning selective coverage of UDP checksum).

Maguire
maguire@it.kth.se

Header Checksum
2006.03.13

IP, ICMP, and Tools 138 of 152
Internetworking/Internetteknik

IPv4 Options

- IPv4 options were intended for network testing & debugging
- Options are variable sized and follow the fixed header
- Contiguous (i.e., no separators)
- Not required fields, but all IP implementations **must** include processing of options
  - Unfortunately, many implementations do not!
- Maxium of 40 bytes available $\Rightarrow$ very limited use
  - Since the maximum header length is 60 bytes and the fixed part is 20 bytes - there is very little space left!

Maguire
maguire@it.kth.se

IPv4 Options
2006.03.13

IP, ICMP, and Tools 139 of 152
Internetworking/Internetteknik

IP Options Encoding

Two styles:

- ## Single byte (only code)

- ## Multiple byte

Option Code: 1 byte

- Copy (to fragments) (1 bit)
    - 0: copy only to the first fragment
    - 1: copy the option to all fragements
- Class (2 bits)
    - 0 (00): Datagram or network control
    - 2 (10): Debugging and measurement
    - 1 (01) and 3 (11) reserved
- Option Number (5 bits)

Option Length: 1 byte, defines total length of option

Data: option specific

Code	Length	Data

copy	class	option number

Maguire
maguire@it.kth.se

IP Options Encoding
2006.03.13

IP, ICMP, and Tools 140 of 152
Internetworking/Internetteknik
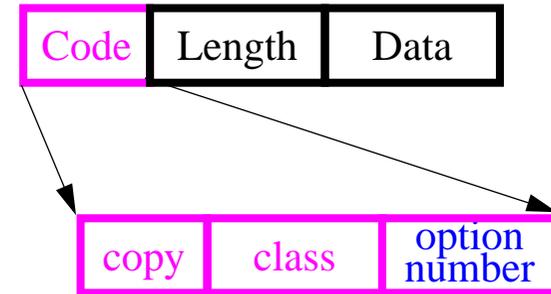
Categories of IP Options

- ## Single byte (only code)
  - No operation (Option Number=0)
  - End of operation (Option Number=1)

- ## Multiple byte
  - Loose Source Route (Option Number=3)
    - Path includes these router, but there can be multiple hops between the specified addresses
  - Time stamp (Option Number=4)
    - Like record route (below), but adds a timestamp at each of the routers (upto the space available - after this an overflow field is incremented - but it is only 4 bits)
  - Record Route (Option Number=7)
  - Strict Source Route (Option Number=9)
    - The exact path is specified

However, due to the very limited space available for the options - these options are of little practical value in todays internet. (Consider the diameter of today's internet versus the number of IP addresses or timestamps that could be in the options field; i.e., record route can only store 9 IP addresses!)

Maguire
maguire@it.kth.se

Categories of IP Options
2006.03.13

IP, ICMP, and Tools 141 of 152
Internetworking/Internetteknik

Internet Control Message Protocol (ICMP)

ICMP [21] is part of the same level as IP, but uses IP for transfers! ICMP is used by layer 3 entities to communicate with each other.

- ICMP PDU: type (8 bits); code (8 bits); checksum (16 bits); parameters (n*32 bits); information (variable length)
  for errors: the information field always includes the first 64 bits of the data field of the original datagram which caused the ICMP message

- ICMP messages include:
  - Destination Unreachable (Network/Host/Protocol/Port/…)
  - Time Exceeded (TTL expired)
  - Parameter problem - IP header error
  - Source Quench (requests source to decrease its data rate)
  - Redirect - tell source to send its messages to a "better address"
  - Echo Request/ Echo reply - for testing (e.g., "ping" program sends an Echo request)
  - Timestamp Request/ Timestamp reply
  - Information Request / Information reply
  - Address Mask Request / Reply
  - Traceroute
  - Datagram conversion error
  - Mobile Host Redirect/Registration Request/Registration Reply
  - IPv6 Where-Are-You/I-Am-Here

Maguire
maguire@it.kth.se

Internet Control Message Protocol (ICMP)
2006.03.13

IP, ICMP, and Tools 142 of 152
Internetworking/Internetteknik

ICMP Port Unreachable Error

Example: (Stevens, Vol. 1, Section 6.5, pp. 77-78)

```
bsdi% tftp
tftp> connect svr4 888    specify host and port number
tftp> get temp.foo        try to fetch a file
Transfer times out.       about 25s later
tftp> quit
```

tcpdump output

1	0.0		arp who-has svr4 tell bsdi
2	0.002050	(0.0020)	arp reply svr4 is-at 0:0:c0:c2:9b:26
3	0.002723	(0.0007)	bsdi.2924 > svr4.8888 udp 20
4	0.006399	(0.0037)	svr4 > bsdi: icmp: svr4 udp port 8888 unreachable
5	5.000776	(4.9944)	bsdi.2924 > svr4.8888 udp 20
6	5.004304	(0.0035)	svr4 > bsdi: icmp: svr4 udp port 8888 unreachable
…			repeats every 5 seconds
11	20.001177	(4.9966)	bsdi.2924 > svr4.8888 udp 20
12	20.004759	(0.0036)	svr4 > bsdi: icmp: svr4 udp port 8888 unreachable

Maguire
maguire@it.kth.se

ICMP Port Unreachable Error
2006.03.13

IP, ICMP, and Tools 143 of 152
Internetworking/Internetteknik

ICMP Redirect

ICMP Redirect message is sent by a router (R1) to the sender of an IP datagram (host) when the datagram should have been sent to a different router (R2).

Maguire
maguire@it.kth.se

ICMP Redirect
2006.03.13

IP, ICMP, and Tools 144 of 152
Internetworking/Internetteknik
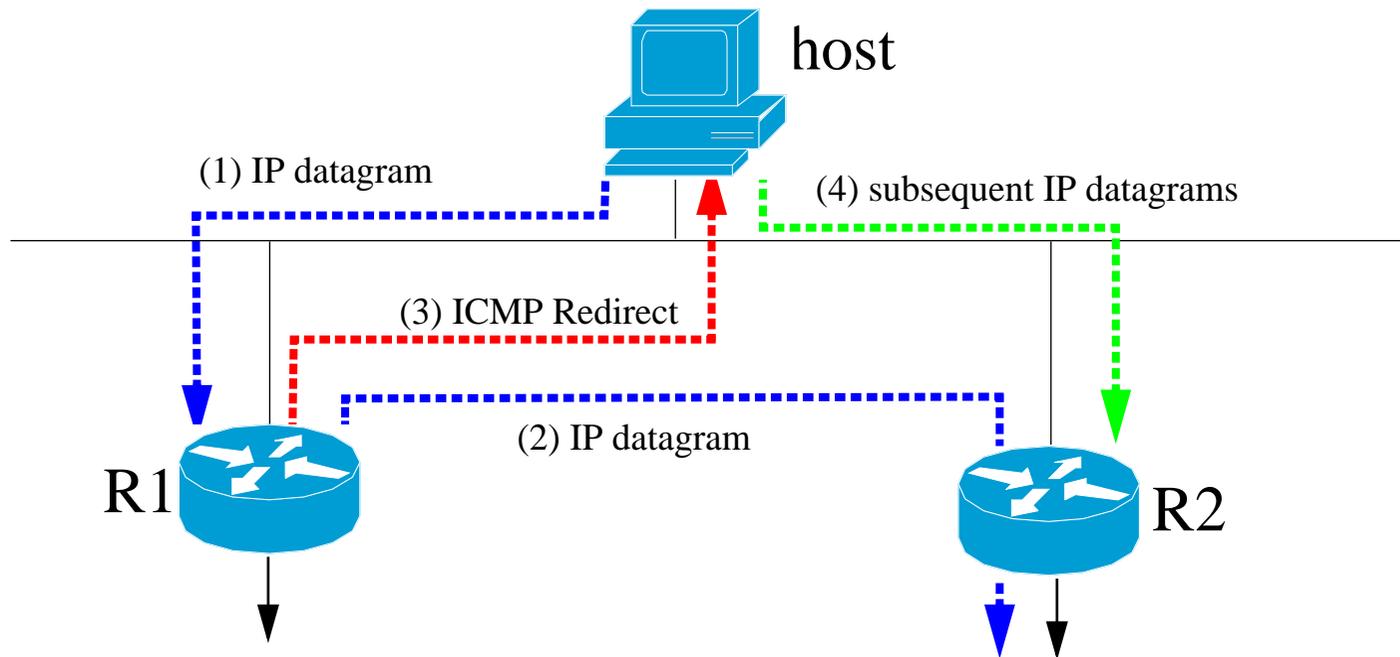
PING: Packet InterNet Groper or sonar echo

Ping was written by Mike Muuss[1] to test host reach-ability. Uses ICMP,most IP implementations support Ping server. Sends an ICMP echo request to a host

Format of ICMP message for Echo request/reply (see Stevens, Vol. 1, figure 7.1, pg. 86)

Type (0 or 8)	code (0)	16 bit checksum
16 bit identifier		16 bit sequence number
Optional data		

Look at ping across different connections[2]:

- LAN

- WAN

- Hardwired SLIP

- Dialup SLIP - extra delay due to the modems and the correction/compression

With IP record route (RR) option tracing the route of the ping datagram.

---

1. Mike Muuss was killed in an automobile accident on November 20, 2000. `http://ftp.arl.mil/~mike/`

2. For examples, see Stevens, Vol. 1, Chapter 7, pp. 86-90.

Maguire
maguire@it.kth.se

PING: Packet InterNet Groper or sonar echo
2006.03.13

IP, ICMP, and Tools 145 of 152
Internetworking/Internetteknik

PING examples

On a Solaris machine:

bash-2.03$ /usr/sbin/ping cyklop.nada.kth.se   *from a machine at IMIT*

cyklop.nada.kth.se is alive

bash-2.03$ /usr/sbin/ping -s cyklop.nada.kth.se

PING cyklop.nada.kth.se: 56 data bytes

64 bytes from cyklop.nada.kth.se (130.237.222.71): icmp_seq=0. time=3. ms

64 bytes from cyklop.nada.kth.se (130.237.222.71): icmp_seq=1. time=1. ms

64 bytes from cyklop.nada.kth.se (130.237.222.71): icmp_seq=2. time=1. ms

^C

----cyklop.nada.kth.se PING Statistics----

3 packets transmitted, 3 packets received, 0% packet loss

round-trip (ms) min/avg/max = 1/1/3

Why did the first ping take longer?

Maguire
maguire@it.kth.se

PING examples
2006.03.13

IP, ICMP, and Tools 146 of 152
Internetworking/Internetteknik

On a HP-UX 11.0 machine:

ping -ov www.kth.se   *from a machine on Telia's ADSL network*

PING www.kth.se: 64 byte packets

64 bytes from 130.237.32.51: icmp_seq=0. time=54. ms

64 bytes from 130.237.32.51: icmp_seq=1. time=38. ms

64 bytes from 130.237.32.51: icmp_seq=2. time=11. ms

64 bytes from 130.237.32.51: icmp_seq=3. time=11. ms

64 bytes from 130.237.32.51: icmp_seq=4. time=11. ms

^C

----www.kth.se PING Statistics----

5 packets transmitted, 5 packets received, 0% packet loss

round-trip (ms) min/avg/max = 11/25/54

5 packets sent via: this is based on the record route information (caused by "-ov")

    217.208.194.247 - fls31o268.telia.com

    213.64.62.150   - fre-d4-geth6-0.se.telia.net

    213.64.62.154   - fre-c3-geth6-0.se.telia.net

    195.67.220.1   - fre-b1-pos0-1.se.telia.net

    130.242.94.4   - STK-PR-2-SRP5.sunet.se

    130.242.204.130 - STK-BB-2-POS4-3.sunet.se

    130.242.204.121 - stockholm-1-FE1-1-0.sunet.se

    130.237.32.3   - [name lookup failed]

    130.237.32.51   - oberon.admin.kth.se

Ping with record route option

$ ping -R www.kth.se

```
PING www.kth.se (130.237.32.51) 56(124) bytes of data.
64 bytes from oberon.admin.kth.se (130.237.32.51): icmp_seq=1
ttl=253 time=2.50ms
RR:      ccsser2 (130.237.15.248)
         ke4-ea4-p2p.gw.kth.se (130.237.211.50)
         kthlan-gw-32-2.admin.kth.se (130.237.32.2)
         oberon.admin.kth.se (130.237.32.51)
         oberon.admin.kth.se (130.237.32.51)
         ea4-ke4-p2p.gw.kth.se (130.237.211.49)
         130.237.15.194
         ccsser2 (130.237.15.248)

64 bytes from oberon.admin.kth.se (130.237.32.51): icmp_seq=2
ttl=253 time=1.73ms        (same route)
64 bytes from oberon.admin.kth.se (130.237.32.51): icmp_seq=3
ttl=253 time=1.80ms        (same route)
64 bytes from oberon.admin.kth.se (130.237.32.51): icmp_seq=4
ttl=253 time=1.90ms        (same route)
```

Maguire
maguire@it.kth.se

Ping with record route option
2006.03.13

IP, ICMP, and Tools 148 of 152
Internetworking/Internetteknik

Useful Tool: Traceroute Programs

Developed by Van Jacobson to see the route that IP datagrams follow from one host to another. Traceroute uses ICMP, TTL field, and an *unreachable UDP port*.

```
svr % traceroute slip
traceroute to slip (140.252.13.65), 30 hops max, 40 byte packets
1 bsdi (140.252.13.35) 20 ms 10 ms 10 ms          20 ms due to ARP
2 slip (140.252.13.65) 120 ms 120 ms 120 ms
```

tcpdump output

1	0.0		arp who-has bsdi tell svr4
2	0.000586	(0.0006)	arp reply bsdi is-at 0:0:c0:6f:2d:40
3	0.003067	(0.0025)	svr4.42804 > slip.33435 udp 12 [ttl 1]
4	0.004325	(0.0013)	bsdi > svr4: icmp: time exceeded in-transit
5	0.069810	(0.0655)	svr4.42804 > slip.33436 udp 12 [ttl 1]
6	0.071149	(0.0013)	bsdi > svr4: icmp: time exceeded in-transit
7	0.085162	(0.0140)	svr4.42804 > slip.33437 udp 12 [ttl 1]
8	0.086375	(0.0012)	bsdi > svr4: icmp: time exceeded in-transit
9	0.118608	(0.0322)	svr4.42804 > slip.33438 udp 12 *ttl=2*
10	0.226464	(0.1079)	slip > svr4: icmp: slip udp port 33438 unreachable
11	0.287296	(0.0608)	svr4.42804 > slip.33439 udp 12 *ttl=2*
12	0.395230	(0.1079)	slip > svr4: icmp: slip udp port 33439 unreachable
13	0.409504	(0.0608)	svr4.42804 > slip.33440 udp 12 *ttl=2*
14	0.517430	(0.1079)	slip > svr4: icmp: slip udp port 33440 unreachable

Maguire
maguire@it.kth.se

Useful Tool: Traceroute Programs
2006.03.13

IP, ICMP, and Tools 149 of 152
Internetworking/Internetteknik

ICMP Summary

- Destination (Network/Host/Protocol/Port/...) Unreachable
- Time Exceeded - i.e., TTL expired
  - Used to implement traceroute
- Parameter problem - IP header error
- Source Quench- asks source to decrease its sending rate
- Redirect - tells the source to send packets to a "better" address
- Echo Request/Echo reply - for testing
  - ping: sends an Echo Request, then measures the time until the matching reply is received
- Timestamp Request/Reply
  - Round Trip Time (RTT) computation
  - Clock synchronization
- Address Mask Request/Reply
  - Allows diskless systems to learn their subnet mask
- Router Solicitation and Advertisment
  - Hosts query routers
  - Routers advertise presence and routes

The above is a partial summary of ICMP's uses.

Maguire
maguire@it.kth.se

ICMP Summary
2006.03.13

IP, ICMP, and Tools 150 of 152
Internetworking/Internetteknik

Summary

This lecture we have discussed:

- IP
- ICMP
- tools: Ping, Traceroute

Maguire
maguire@it.kth.se

Summary
2006.03.13

IP, ICMP, and Tools 151 of 152
Internetworking/Internetteknik

References

[19] K. Ramakrishnan, S. Floyd, and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", IETF RFC 3168, September 2001.

[20] J. Mogul and S. Deering, "Path MTU Discovery", IETF, RFC 1191, November 1990.

[21] J. Postel, Internet Control Message Protocol, IETF, RFC 792, September 1981 `http://www.ietf.org/rfc/rfc0792.txt`

Maguire
maguire@it.kth.se

References
2006.03.13

IP, ICMP, and Tools 152 of 152
Internetworking/Internetteknik

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 4: UDP and friends

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapters 11, 16, 17

KTH Information and Communication Technology

Outline

- UDP
- Socket API
- BOOTP
- DHCP
- DNS, DDNS

Maguire
maguire@it.kth.se

Outline
2006.03.13

UDP and friends 154 of 220
Internetworking/Internetteknik

Transport layer protocols

The transport layer is responsible for end-to-end delivery of entire messages

- uses Protocol Port and/or Port Number to demultiplex the incoming packet so that it can be delivered to a specific process
- Segmentation and Reassembly
  - Divides a message into transmittable segments and reassemble them at the receive
- Connection control - for connection-oriented transport protocols
- End-to-end Flow Control (in contrast to link level flow control)
- End-to-end Error Control (in contrast to link level error control)

Maguire
maguire@it.kth.se

Transport layer protocols
2006.03.13

UDP and friends 155 of 220
Internetworking/Internetteknik

Main Transport layer protocols

Three main transport layer protocols:

- User Datagram Protcol (UDP) <<< today's topic
  - Connectionless unreliable service

- Transmission Control Protocol (TCP)
  - Connection-oriented reliable stream service

- Stream Control Transmission Protocol (STCP)
  - a modern transmission protocol with many facilities which the user can chose from

Maguire
maguire@it.kth.se

Main Transport layer protocols
2006.03.13

UDP and friends 156 of 220
Internetworking/Internetteknik

User Datagram Protocol (UDP)

- Datagram-oriented transport layer protocol
- Provides connectionless unreliable service
- No reliability guarantee
- Checksum covers both header and data, end-to-end, but optional
  - if you care about your data you should be doing end-to-end checksums or using an even stronger error detection (e.g., MD5).
- An UDP datagram is silently discarded if checksum is in error.
  - No error message is generated
- Lots of UDP traffic is only sent locally
  - thus the reliability is comparable to the error rate on the local links. (see Stevens, Vol. 1, figure 11.5, pg. 147 for comparison of Ethernet, IP, UDP, and TCP checksum errors)
- Each output operation results in one UDP datagram, which causes one IP datagram to be sent
- Applications which use UDP: DNS, TFTP, BOOTP, DHCP, SNMP, NFS, VoIP, etc.
  - An advantage of UDP is that it is a base to build your own protocols on
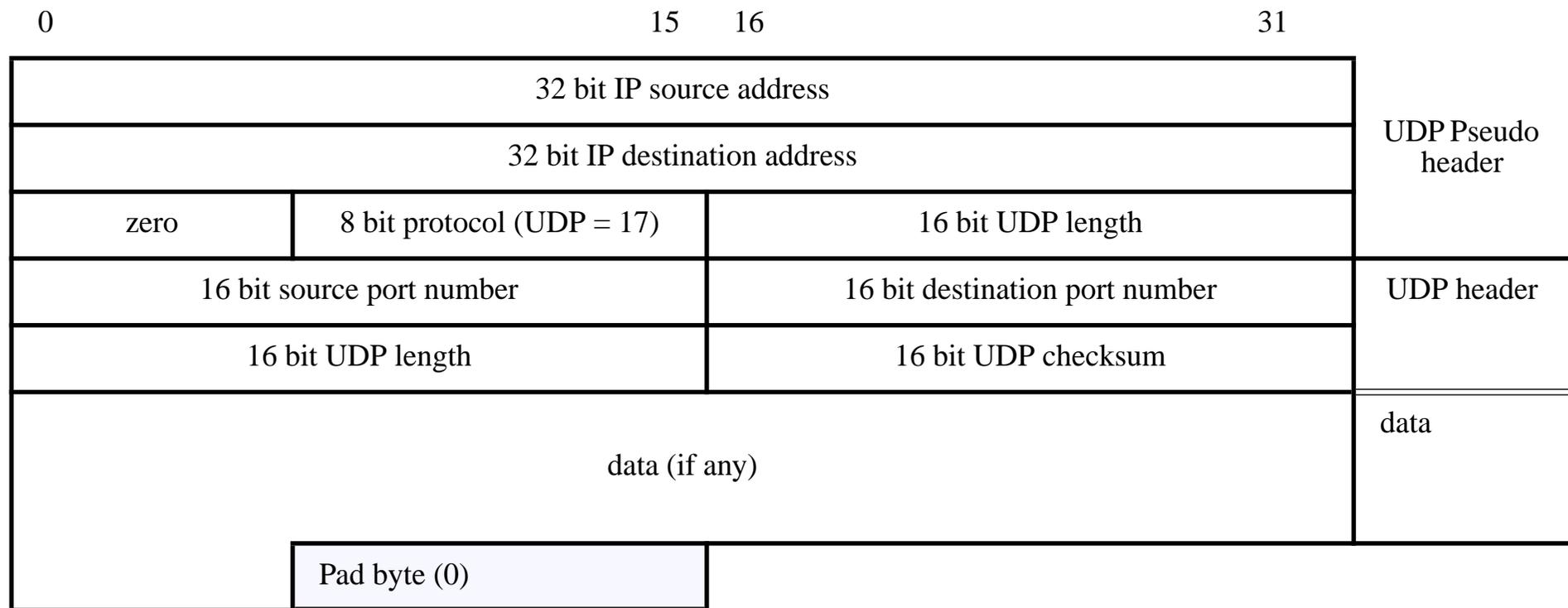 - Especially if you don't need reliability and in order delivery of lots of data

Maguire
maguire@it.kth.se

User Datagram Protocol (UDP)
2006.03.13

UDP and friends 157 of 220
Internetworking/Internetteknik

UDP Header

8 byte header + possible data

| 0 | 15 | 16 | 31 |

16 bit source port number	16 bit destination port number
16 bit UDP length	16 bit UDP checksum
data (if any)	

IP datagram

UDP datagram

IP header	UDP header	UDP data
20 bytes	8 bytes	$0 .. 2^{16}-(20+8+1) = 65{,}507$ bytes

Port 1	Port 2	Port n

demux based on port number

Maguire
maguire@it.kth.se

UDP Header
2006.03.13

UDP and friends 158 of 220
Internetworking/Internetteknik
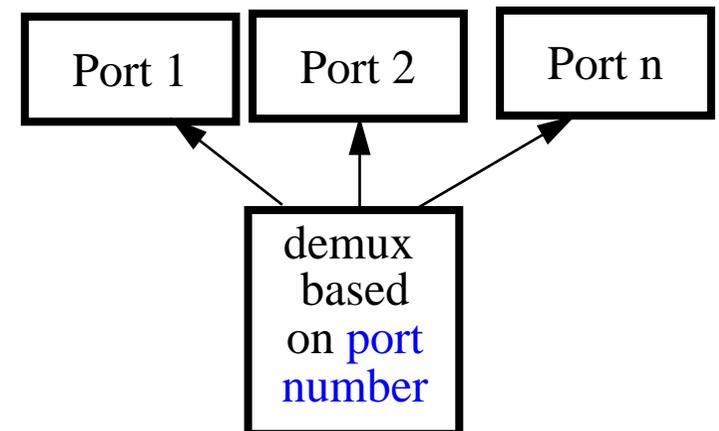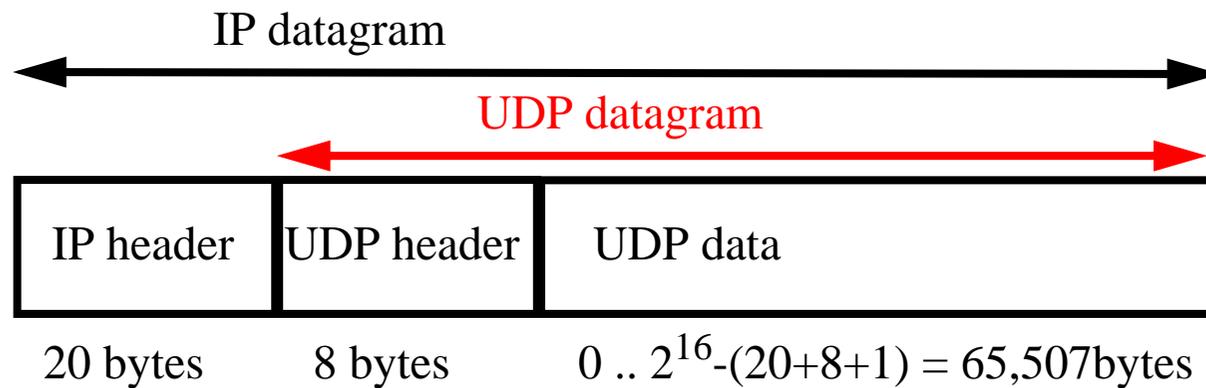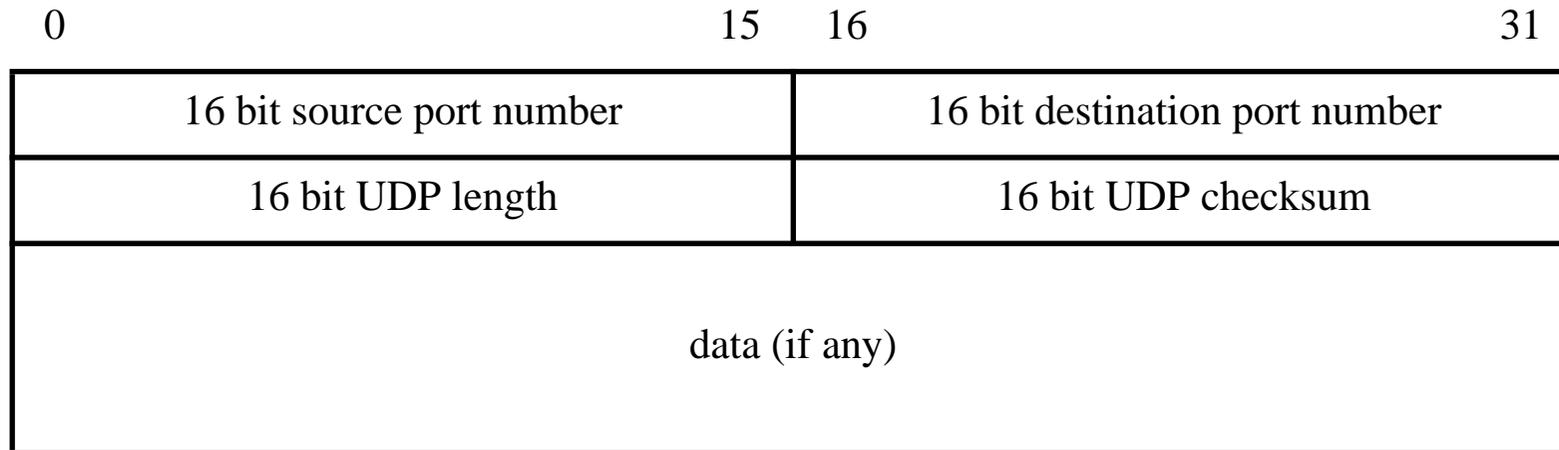
UDP Checksum and Pseudo-Header

- UDP checksum covers more info than is present in the UDP datagram alone: pseudo-header and pad byte (0) {to even number of 16 bit words}.

- Propose: to verify the UDP datagram reached its correct destination: right port number at the right IP address.

- Pseudo-header and pad byte are not transmitted with the UDP datagram, only used for checksum computation.

0	15	16	31	
32 bit IP source address				UDP Pseudo header
32 bit IP destination address				
zero	8 bit protocol (UDP = 17)	16 bit UDP length		
16 bit source port number		16 bit destination port number		UDP header
16 bit UDP length		16 bit UDP checksum		
data (if any)				data
Pad byte (0)				

Maguire
maguire@it.kth.se

UDP Checksum and Pseudo-Header
2006.03.13

UDP and friends 159 of 220
Internetworking/Internetteknik

Reserved and Available UDP Port Numbers

	keyword	UNIX keyword	Description
0			reserved
7	ECHO	echo	Echo
9	DISCARD	discard	Discard == sink null
11	USERS	systat	Active users
13	DAYTIME	daytime	Daytime
15	-	netstat	Network status program
17	QUOTE	qotd	Quote of the day
19	CHARGEN	chargen	Character generator
37	TIME	time	Time server
39	RLP	rlp	Resource Location Protocol
42	NAMESERVER	name	Host Name Server
43	NICNAME	whois	Who is
53	DOMAIN	domain	Domain Name Server
67	BOOTPS	bootps	Bootstrap Protocol Server
68	BOOTPC	bootpc	Bootstrap Protocol Client
69	TFTP	tftp	Trivial File Transfer Protocol
88	KERBEROS	kerberos5	Kerberos v5 kdc
111	SUNRPC	sunrpc	SUN Remote Procedure Call (portmap)
123	NTP	ntp	Network Time Protocol
137	netbios_ns	netbios_ns	NetBIOS name service
138	netbios_dgm	netbios_dgm	NetBIOS Datagram Service
139	netbios_ssn	netbios_ssn	NetBIOS Session Service
161		snmp	Simple Network Management Protocol Agent
162		snmp-trap	Simple Network Management Protocol Traps
512		biff	mail notification
513		who	remote who and uptime
514		syslog	remote system logging
517		talk	conversation
518		ntalk	new talk, conversation
520		route	routing information protocol
525		timed	remote clock synchronization
533	netwall	netwall	Emergency broadcasting
750	kerberos	kerberos	Kerberos (server)
6000 + display number			X11 server
7000			X11 font server

Maguire
maguire@it.kth.se
Reserved and Available UDP Port Numbers
2006.03.13
UDP and friends 160 of 220
Internetworking/Internetteknik

Port numbers in three groups

Range	Purpose
0 .. 1023	System (Well-Known) Ports[a]
1024 .. 49151	User (Registered) Ports
49152 .. 65535	Dynamic and/or Private Ports

a. Roughly 300 well know port numbers remain unassigned and 38 reserved
Roughly 26k registered port numbers remain unassigned and 9 reserved

'For the purpose of providing services to unknown callers, a service contact port is defined. This list specifies the port used by the server process as its contact port. The contact port is sometimes called the "well-known port".'

http://www.iana.org/assignments/port-numbers

Linux chooses the local port to use for TCP and UDP traffic from this range:
```
$ cat /proc/sys/net/ipv4/ip_local_port_range
  1024    29999
```

Maguire
maguire@it.kth.se

Port numbers in three groups
2006.03.13

UDP and friends 161 of 220
Internetworking/Internetteknik

MTU and Datagram Fragmentation

If datagram size > MTU, perform fragmentation.

- At sending host or at intermediate router (IPv4).
- Reassembled only at final destination.

Example: 1501 (20 + 8 + 1473 data) on Ethernet (MTU=1500):
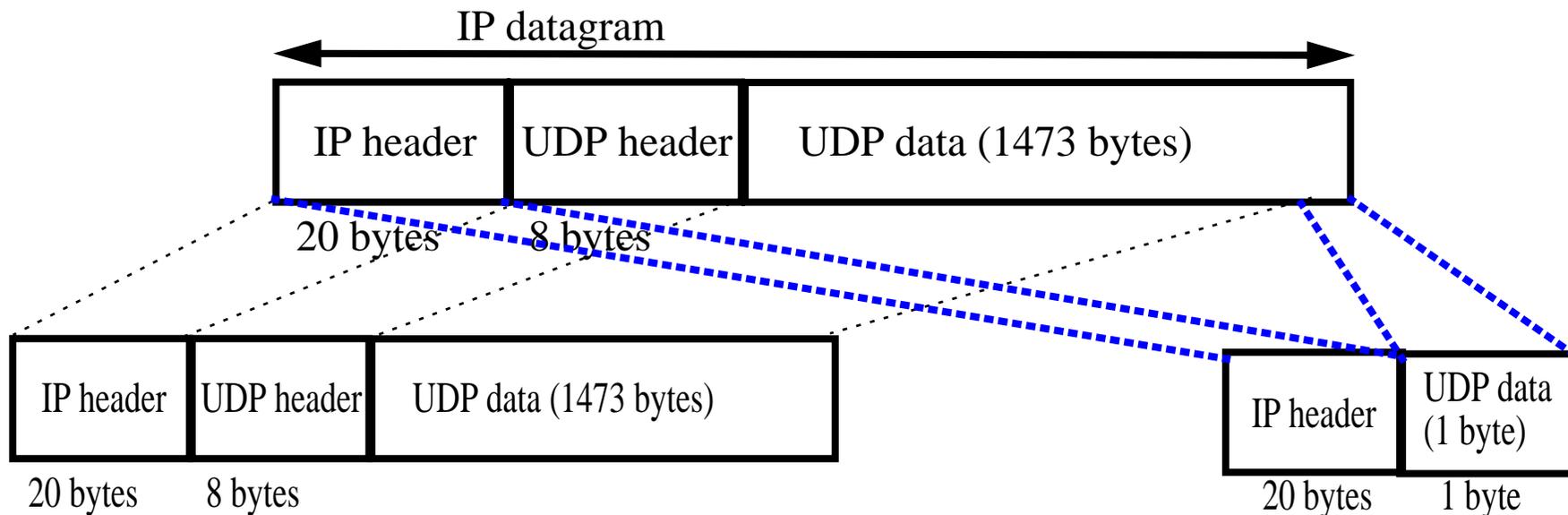
- Note there is no UDP header in the second fragment.
- Therefore, a frequent operation is to compute the path MTU before sending anything else. (see RFC 1191 for the table of common MTUs)

Maguire
maguire@it.kth.se

MTU and Datagram Fragmentation
2006.03.13

UDP and friends 162 of 220
Internetworking/Internetteknik

Fragmentation Required

If datagram size > MTU, DF (Don't Fragment) in IP header is on, then the router sends ICMP Unreachable Error.

Of course this can be used to find Path MTU.

Maguire
maguire@it.kth.se

Fragmentation Required
2006.03.13

UDP and friends 163 of 220
Internetworking/Internetteknik

Interaction between UDP and ARP

With ARP cache empty, send a UDP datagram with 8192 bytes onto an Ethernet

- 8192 bytes > ethernet MTU, therefore 6 fragments are created by IP
- if ARP cache is empty, first fragment causes ARP request to be sent
- This leads to two timing questions:

1. Are the remaining fragments sent before the ARP reply is received?

2. What does ARP do with multiple packets to the same destination while waiting for a reply?

Example under BSD

```
Bsdi% arp -a        ARP cache is empty
Bsdi% sock -u -i -n1 -w8192 svr4 discard
              10.0                    arp who-has svr4 tell bsdi
              20.001234  (0.0012)     arp who-has svr4 tell bsdi
              30.001941  (0.0007)     arp who-has svr4 tell bsdi
              40.002775  (0.0008)     arp who-has svr4 tell bsdi
              50.003495  (0.0007)     arp who-has svr4 tell bsdi
              60.004319  (0.0008)     arp who-has svr4 tell bsdi
              70.008772  (0.0045)     arp reply svr4 is-at 0:0:c0:c2:9b:26
              80.009911  (0.0011)     arp reply svr4 is-at 0:0:c0:c2:9b:26
              90.011127  (0.0012)     bsdi > svr4: (frag 10863:800@7400)
             100.011255  (0.0001)     arp reply svr4 is-at 0:0:c0:c2:9b:26
             110.012562  (0.0013)     arp reply svr4 is-at 0:0:c0:c2:9b:26
             120.013458  (0.0009)     arp reply svr4 is-at 0:0:c0:c2:9b:26
             130.014526  (0.0011)     arp reply svr4 is-at 0:0:c0:c2:9b:26
             140.015583  (0.0011)     arp reply svr4 is-at 0:0:c0:c2:9b:26
```

Maguire
maguire@it.kth.se

Interaction between UDP and ARP
2006.03.13

UDP and friends 164 of 220
Internetworking/Internetteknik

- on a BSDI system:
  - each of the additional (5) fragments caused an ARP request to be generated
    - this **violates** the Host Requirements RFC - which tries to prevent ARP flooding by limiting the maximum rate to 1 per second
  - when the ARP reply is received the **last** fragment is sent
    - Host Requirements RFC says that ARP should save at least one packet and this should be the latest packet
  - unexplained anomaly: the System Vr4 system sent 7 ARP replies back!
  - no ICMP "time exceeded during reassembly" message is sent
    - BSD derived systems - never generate this error!
      It does set the timer internally and discard the fragments, but never sends an ICMP error.
    - fragment 0 (which contains the UDP header) was not received - so there is no way to know which process sent the fragment; thus unless fragment 0 is received - you are not required to send an ICMP "time exceeded during reassembly" error.

Not just a fluke (i.e., a rare event)

- The same error occurs even if you don't have fragmentation - simply sending multiple UDP datagrams rapidly when there is no ARP entry is sufficient!

- NFS sends UDP datagrams whose length just exceeds 8192 bytes
  - NFS will timeout and resend
  - however, there will always be this behavior - if the ARP cache has no entry for this destination!

Maguire
maguire@it.kth.se

Interaction between UDP and ARP
2006.03.13

UDP and friends 165 of 220
Internetworking/Internetteknik

Still a problem?

A UDP with 8192 payload to echo port as seen on SuSE 9.2 linux 2.6.8-24:

```
No.      Time            Source                  Destination           Protocol Info
    37 3.020002     172.16.33.16            Broadcast             ARP       Who has 172.16.33.5?  Tell 172.16.33.16
    38 3.021385     172.16.33.5             172.16.33.16          ARP       172.16.33.5 is at 00:40:8c:24:37:f4
    39 3.021422     172.16.33.16            172.16.33.5           IP        Fragmented IP protocol (proto=UDP 0x11, off=4440)
    40 3.021452     172.16.33.16            172.16.33.5           IP        Fragmented IP protocol (proto=UDP 0x11, off=5920)
    41 3.021480     172.16.33.16            172.16.33.5           IP        Fragmented IP protocol (proto=UDP 0x11, off=7400)
```

3.021385-3.020002=.001383 sec. $\Rightarrow$ 1.383ms for the ARP reply

All but the last 3 fragments are dropped! Including the initial echo request packet -- so in the fragments that do arrive you don't know who they are for -- because the first fragment was lost!

With an even larger UDP packet

I removed the arp cache entry with: /sbin/arp -i eth1 -d 172.16.33.5

When sending 65500 bytes of UDP payload -- it looses many packets (in fact all but the last 3 fragments)!!!

```
No.    Time       Source              Destination          Protocol Info
   36 4.342158    172.16.33.16        Broadcast            ARP      Who has 172.16.33.5?  Tell 172.16.33.16
   37 4.342875    172.16.33.5         172.16.33.16         ARP      172.16.33.5 is at 00:40:8c:24:37:f4
   38 4.342906    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=62160)
   39 4.342932    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=63640)
   40 4.342986    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=65120)
```

With the entry in the ARP cache get:

```
No.    Time       Source              Destination          Protocol Info
   35 5.118063    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=1480)
   36 5.118095    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=2960)
   37 5.118115    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=4440)
   38 5.118214    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=5920)
   39 5.118328    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=7400)
   40 5.118450    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=8880)
   41 5.118574    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=10360)
   42 5.118695    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=11840)
   43 5.118819    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=13320)
          ….
   72 5.122385    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=56240)
   73 5.122508    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=57720)
   74 5.122631    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=59200)
   75 5.122787    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=60680)
   76 5.122877    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=62160)
   77 5.122999    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=63640)
   78 5.123122    172.16.33.16        172.16.33.5          IP       Fragmented IP protocol (proto=UDP 0x11, off=65120)
```

The initial UDP Echo request is still lost! The key parameter is
`/proc/sys/net/ipv4/neigh/ethX/unres_qlen` where X is the
interface (i.e., eth0, eth1, …) -- the default value is 3.

Maguire
maguire@it.kth.se

With an even larger UDP packet
2006.03.13

UDP and friends 167 of 220
Internetworking/Internetteknik

Maximum UDP Datagram size

- theoretical limit: 65,535 bytes - due to (IP's) 16-bit total length field
  - with 20 bytes of IP header + 8 bytes of UDP header $\Rightarrow$ 65,507 bytes of user data
- two limits:
  - sockets API limites size of send and receive buffer; generally 8 kbytes, but you can call a routine to change this
  - TCP/IP implementation - Stevens found various limits to the sizes - even with loopback interface (see Stevens, Vol. 1, pg. 159)
- Hosts are required to handle at least 576 byte IP datagrams, thus lots of protocols limit themselves to 512 bytes or less of data:
  - DNS, TFTP, BOOTP, and SNMP

Maguire
maguire@it.kth.se

Maximum UDP Datagram size
2006.03.13

UDP and friends 168 of 220
Internetworking/Internetteknik

Datagram truncation

What if the application is not prepared to read the datagram of the size sent?

Implementation dependent:

- traditional Berkeley: silently truncate
- 4.3BSD and Reno: **can** notify the application that the data was truncated
- SVR4: excess data returned in subsequent reads - application is not told that this all comes from one datagram
- TLI: sets a flag that more data is available, subsequent reads return the rest of the datagram

Maguire
maguire@it.kth.se

Datagram truncation
2006.03.13

UDP and friends 169 of 220
Internetworking/Internetteknik

Socket API

- int socket(int domain, int type, int protocol);
  - creates an endpoint description that you can use to send and receive network traffic
- int bind(int sockfd, struct sockaddr *my_addr, socklen_t addrlen);
  - binds a socket to the local address and port: my_address
- int connect(int sockfd, const struct sockaddr *serv_addr, socklen_t addrlen);
  - connect the local socket to a remote socket
- int listen(int s, int backlog);
  - indicates that socket is willing to accept connections & limites the queue of pending connections
- int accept(int s, struct sockaddr *addr, socklen_t *addrlen);
  - accepted the first connection request from the queue and connects it to the socket
- connection oriented sending and receiving:
  - ssize_t send(int s, const void *buf, size_t len, int flags);
  - ssize_t recv(int s, void *buf, size_t len, int flags);
- datagram sending and receiving:
  - ssize_t sendto(int s, const void *buf, size_t len, int flags, const struct sockaddr *to, socklen_t tolen);
  - ssize_t recvfrom(int s, void *buf, size_t len, int flags, struct sockaddr *from, socklen_t *fromlen);
- int close(int fd) and int shutdown(int s, int how) - end it all!

Maguire
maguire@it.kth.se

Socket API
2006.03.13

UDP and friends 170 of 220
Internetworking/Internetteknik

Learning about Socket programming

For examples of using the socket API and networking programming see :

- Richard Stevens, *UNIX Network Programming, Volume 1, Second Edition: Networking APIs: Sockets and XTI*, Prentice Hall, 1998, ISBN 0-13-490012-X
  - source code `ftp://ftp.kohala.com/pub/rstevens/unpv12e.tar.gz`
  - errata list: `http://www.kohala.com/~rstevens/typos.unpv12e.txt`
- Brian "Beej" Hall, "Beej's Guide to Network Programming: Using Internet Sockets", 04/08/2004 07:22:02 PM

  `http://www.ecst.csuchico.edu/~beej/guide/net/`

Two addition socket functions for controlling various properties of sockets are:

- int getsockopt(int s, int level, int optname,   void *optval, socklen_t *optlen);
- int setsockopt(int s, int level, int optname, const void *optval, socklen_t optlen);

For example, the options `SO_SNDBUF` and `SO_RCVBUF` - control the size of the sending buffer and the receiver buffer.

Maguire
maguire@it.kth.se

Learning about Socket programming
2006.03.13

UDP and friends 171 of 220
Internetworking/Internetteknik

Simple UDP client

```c
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#define bigBufferSize      8192
#define destination_host "172.16.33.5"

main(argc, argv)
int argc;
char **argv;
{ int client_socket_fd;            /* Socket to client, server */
  struct sockaddr_in server_addr; /* server's address */
  char bigBuffer[bigBufferSize];  /* buffer of data to send as payload */
  int sendto_flags=0;
                                  /* create a UDP socket */
  if ((client_socket_fd = socket(AF_INET, SOCK_DGRAM, IPPROTO_UDP)) == -1) {
    perror("Unable to open socket"); exit(1); };

                                  /* initialize the server address structure */
  memset( (char*)&server_addr, 0, sizeof(server_addr));
  server_addr.sin_family=AF_INET;
  server_addr.sin_port=htons(9);  /* 9 is the UDP port number for Discard */

  if (inet_aton(destination_host, (struct sockaddr*)&server_addr.sin_addr) == 0) {
    fprintf(stderr, "could not get an address for: %s", destination_host);exit(1);}

  if ((sendto(client_socket_fd, bigBuffer, bigBufferSize,
          sendto_flags, (struct sockaddr*)&server_addr, sizeof(server_addr))) == -1) {
    perror("Unable to send to socket"); close(client_socket_fd); exit(1);}

  close(client_socket_fd);         /* close the socket */
  exit(0);
}
```

Maguire
maguire@it.kth.se

Simple UDP client
2006.03.13

UDP and friends 172 of 220
Internetworking/Internetteknik

UDP server design

Stevens, Vol, 1, pp. 162-167 discusses how to program a UDP server

You can often determine what IP address the request was sent to (i.e., the destination address):

- for example: thus ignoring datagrams sent to a broadcast address

You can limit a server to a given incoming IP address:

- thus limiting requests to a given interface

You can limit a server to a given foreign IP address and port:

- only accepting requests from a given foreign IP address and port #

Multiple recipients per port (for implementations with multicasting support)

- setting SO_REUSEADDR socket option $\Rightarrow$ each process gets a copy of the incoming datagram

Note: limited size input queue to each UDP port, can result in silent discards without an ICMP message being sent back (since OS discarded, not the network!)

Maguire
maguire@it.kth.se
Simple UDP client
2006.03.13
UDP and friends 173 of 220
Internetworking/Internetteknik

UDP listener example

```c
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>

#define bigBufferSize 8192
#define my_port        52000
#define destination_host "127.0.0.1"

main(argc, argv)
int argc;
char **argv;
{
  int client_socket_fd;                    /* Socket to client, server */
  struct sockaddr_in client_addr;          /* client's address */
  struct sockaddr_in other_addr;           /* other party's address */
  int other_addr_len;
  char bigBuffer[bigBufferSize];
  int sendto_flags=0;

                                           /* create a UDP socket */
  if ((client_socket_fd = socket(AF_INET, SOCK_DGRAM, IPPROTO_UDP)) == -1) {
    perror("Unable to open socket"); exit(1); };

  memset((char*)&client_addr, 0, sizeof(client_addr)); /* initialize address structure */
  client_addr.sin_family=AF_INET;
  client_addr.sin_port=htons(my_port);
  client_addr.sin_addr.s_addr = htonl(INADDR_ANY);

  if (bind(client_socket_fd, (struct sockaddr*)&client_addr, sizeof(client_addr))==-1) {
    close(client_socket_fd); exit(1); }

  if ((recvfrom(client_socket_fd, bigBuffer, bigBufferSize,
            sendto_flags, (struct sockaddr*)&other_addr, &other_addr_len)) == -1) {
    perror("Unable to receive from socket"); close(client_socket_fd); exit(1); }

  printf("Received packet from %s:%d\nData: %s\nString length=%d\n",
  inet_ntoa(other_addr.sin_addr), ntohs(other_addr.sin_port), bigBuffer, strlen(bigBuffer));

  close(client_socket_fd); exit(0);}
```

Maguire
maguire@it.kth.se

Simple UDP client
2006.03.13

UDP and friends 174 of 220
Internetworking/Internetteknik

Changed the client

Changing the following:

```
#define destination_host "127.0.0.1"
#define my_port          52000

server_addr.sin_port=htons(my_port);
```

Adding some content to the bigBuffer:

```
sprintf(bigBuffer, "This is a simple test string to be sent to the
other party\n");
```

Sending only as much of the buffer as necessary:

```
if ((sendto(client_socket_fd, bigBuffer, strlen(bigBuffer),
            sendto_flags, (struct sockaddr*)&server_addr,
sizeof(server_addr))) == -1) {…}
```

Results in the listener outputting:

```
Received packet from 127.0.0.1:1260
Data: This is a simple test string to be sent to the other party

String length=59
```

Maguire
maguire@it.kth.se

Simple UDP client
2006.03.13

UDP and friends 175 of 220
Internetworking/Internetteknik

Building a UDP packet from scratch

```c
/*  simple example of building a UDP packet from scratch, based on the program:
    PingPong - 970621 by Willy TARREAU  <tarreau@aemiaif.ibp.fr>
      http://www.insecure.org/sploits/inetd.internal_udp_ports.DOS.attack.html
  As this program uses RAW sockets, you must be root to run it
*/


#include <netinet/in.h>
#include <netinet/ip.h>
#include <netinet/udp.h>
#include <netdb.h>
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
#include <ctype.h>
#include <errno.h>

struct sockaddr addrfrom;
struct sockaddr addrto;
int s;
u_char outpack[65536];
struct iphdr *ip;
struct udphdr *udp;

main(int argc, char **argv) {
    struct sockaddr_in *from;
    struct sockaddr_in *to;
    struct protoent *proto;
    int i;
    char *src,*dest;
    int srcp, destp;
    int packetsize,datasize;

    if (argc!=5) {fprintf(stderr,"Usage: %s src_addr src_port dst_addr dst_port\n", argv[0]);
              fprintf(stderr,"src_addr and dst_addr must be given as IP addresses (xxx.xxx.xxx.xxx)\n");
              exit(2);}
    src=argv[1]; srcp=atoi(argv[2]); dest=argv[3]; destp=atoi(argv[4]);
```

Maguire
maguire@it.kth.se

Building a UDP packet from scratch
2006.03.13

UDP and friends 176 of 220
Internetworking/Internetteknik

```c
if (!(proto = getprotobyname("raw"))) {perror("getprotobyname(raw)");exit(2);}

if ((s = socket(AF_INET, SOCK_RAW, proto->p_proto)) < 0) {perror("socket");exit(2);}

memset(&addrfrom, 0, sizeof(struct sockaddr));
from = (struct sockaddr_in *)&addrfrom;
from->sin_family = AF_INET;
from->sin_port=htons(srcp);
if (!inet_aton(src, &from->sin_addr)) {fprintf(stderr,"Incorrect address for 'from': %s\n",src);exit(2); }

memset(&addrto, 0, sizeof(struct sockaddr));
to = (struct sockaddr_in *)&addrto;
to->sin_family = AF_INET;
to->sin_port=htons(destp);
if (!inet_aton(dest, &to->sin_addr)) {fprintf(stderr,"Incorrect address for 'to': %s\n",dest);exit(2); }

packetsize=0;

/* build a UDP packet from scratch */

ip=(struct iphdr *)outpack;
ip->version=4;        /* IPv4 */
ip->ihl=5;            /* IP header length: 5 words  */
ip->tos=0;            /* no special type of service */
ip->id=0;             /* no ID */
ip->frag_off=0;       /* not a fragment - so there is no offset */
ip->ttl=0x40;         /* TTL = 64 */

if (!(proto = getprotobyname("udp"))) {perror("getprotobyname(udp)"); exit(2);}

ip->protocol=proto->p_proto;
ip->check=0;          /* null checksum, will be automatically computed by the kernel */
ip->saddr=from->sin_addr.s_addr; /* set source and destination addresses */
ip->daddr=to->sin_addr.s_addr;
/* end of ip header */
```

```
    packetsize+=ip->ihl<<2;
                                        /* udp header */
    udp=(struct udphdr *)((int)outpack + (int)(ip->ihl<<2));
    udp->source=htons(srcp);
    udp->dest=htons(destp);
    udp->check=0;                       /* ignore UDP checksum */
    packetsize+=sizeof(struct udphdr);
    /* end of udp header */

    /* add data to UDP payload if you want: */
    for (datasize=0;datasize<8;datasize++) {
        outpack[packetsize+datasize]='A'+datasize;
    }
    packetsize+=datasize;
    udp->len=htons(sizeof(struct udphdr)+datasize);
    ip->tot_len=htons(packetsize);

    if (sendto(s, (char *)outpack, packetsize, 0, &addrto, sizeof(struct sockaddr))==-1)
        {perror("sendto"); exit(2);}

    printf("packet sent !\n");
    close(s);
    exit(0);
}
```

ICMP Source Quench Error

Since UDP has no flow control, a node could receive datagrams faster than it can process them. In this situation the host **may** send an ICMP source quench.

Note: "**may** be generated" - it is not required to generate this error

Stevens (Vol. 1, pp. 160-161) gives the example of sending 100 1024-byte datagrams from a machine on an ethernet via a router and SLIP line to another machine:
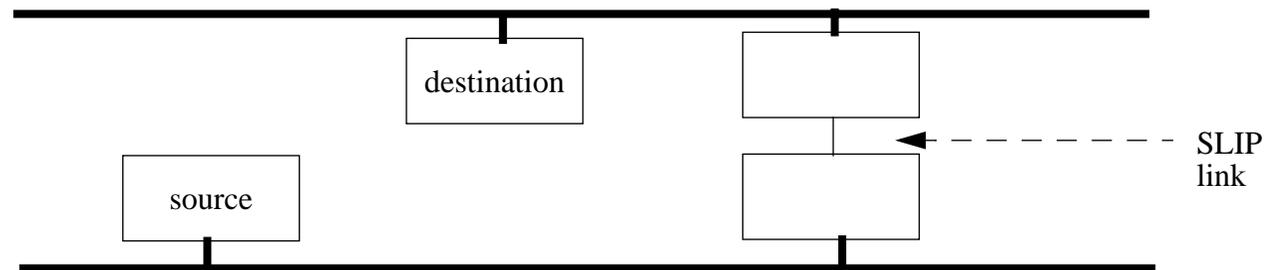
Figure 29: simplified from Stevens, Vol. 1, inside cover

- SLIP link is ~1000 times slower than the ethernet
- 26 datagrams are transmitted, then a source quench is sent for each successive datagram

Maguire
maguire@it.kth.se

Building a UDP packet from scratch
2006.03.13

UDP and friends 179 of 220
Internetworking/Internetteknik

- the router gets all 100 packets, before the first has been sent across the link!
  - the new Router Requirements RFC - says that routers should not generate source quench errors, since it just consumes network bandwidth and it is an ineffective and unfair fix for congestion
- In any case, the sending program never responded to the source quench errors!
  - BSD implementations ignore received source quenchs if the protocol is UDP
  - the program finished before the source quench was received!

Thus if you want reliability you have to build it in and do end-to-end flow control, error checking, and use (and thus wait for) acknowledgements.

Maguire
maguire@it.kth.se

Building a UDP packet from scratch
2006.03.13

UDP and friends 180 of 220
Internetworking/Internetteknik

No error control

Since UDP has no error control, the sender has to take responsibility for sending the datagram again if this datagram must be delivered.

But how does the send know if a datagram was successfully delivered?

- Unless the receiver sends a reply (or does some action due to receiving a given datagram) the sender will **not** know! [loss]
- Note that without some additional mechanism - the sender doesn't know if the datagram was delivered multiple times! [duplicates]

If you want reliability you have to build your own protocol on top of UDP to achieve it. This includes deciding on your own retransmission scheme, timeouts, etc.

Maguire
maguire@it.kth.se

No error control
2006.03.13

UDP and friends 181 of 220
Internetworking/Internetteknik

BOOTP: Bootstrap Protocol (RFC 951)

Although you can figure out who you are, i.e., your IP address, via RARP - many machines want more information.

BOOTP requests and answer are sent via UDP (port 67 server; port 68 client)
- so it is easy to make a user space server
- the client (who wants the answer) need not have a full TCP/IP, it can simply send what **looks** like a UDP datagram with a BOOTP request[1].

Opcode (1=request, 2=reply)	hardware type (1=ethernet)	hardware address length (6 for ethernet)	hop count
transaction ID			
number of seconds		unused	
client IP address			
your IP address			
server IP address			
gateway IP address			
client hardware address (16 bytes of space)			
server hostname (64 bytes)			
Boot file name (128 bytes)			
Vendor specific information (64 bytes)			

---

1. see Stevens, Vol. 1, figure 16.2, pg. 216

Maguire
maguire@it.kth.se

BOOTP: Bootstrap Protocol (RFC 951)
2006.03.13

UDP and friends 182 of 220
Internetworking/Internetteknik

BOOTP continued

When a request is sent as an IP datagram:

- if client does not know its IP address it uses 0.0.0.0
- if it does not know the server's address it uses 255.255.255.255
- if the client does not get a reply, it tries again in about 2 sec.

Maguire
maguire@it.kth.se

BOOTP continued
2006.03.13

UDP and friends 183 of 220
Internetworking/Internetteknik

Vendor specific information (RFC 1497 and RFC1533)

- if this area is used the first 4 bytes are: IP address 99.130.83.99 this is called the "**magic cookie**"

- the rest of the area is a list of items, possibly including:
  - Pad (tag=0);
  - Subnet mask (tag=1);
  - Time offset (tag=2);
  - List of IP addresses of Gateways (tag=3);
  - Time server's IP address (tag=4);
  - Name Server (tag=5);
  - Domain Name Server (tag=6);
  - LOG server (tag=7); …
  - LPR server (tag=9); …
  - this Host's name (tag=12);
  - Boot file size (tag=13); …
  - Domain name (tag=15); …
  - End (tag=255)

Maguire
maguire@it.kth.se

Vendor specific information (RFC 1497 and RFC1533)
2006.03.13

UDP and friends 184 of 220
Internetworking/Internetteknik

DHCP: Dynamic Host Configuration Protocol (RFC 1531)

Extends the Vendor specific options area by 312 bytes.

This protocol is designed to make it easier to allocate (and reallocate) addresses for clients. DHCP defines:

- Requested IP Address - used in client request (DHCPDISCOVER) to request that a particular IP address (tag=50)

- IP Address Lease Time - used in a client request (DHCPDISCOVER or DHCPREQUEST) to request a lease time for the IP address. In a server reply (DHCPOFFER), specific lease time offered. (tag=51)

- Option Overload - used to indicate that the DHCP "sname" or "file" fields are being overloaded by using them to carry DHCP options. A DHCP server inserts this option if the returned parameters will exceed the usual space allotted for options, i.e., it uses the sname and file fields for another purpose! (tag=52)

Maguire
maguire@it.kth.se

DHCP: Dynamic Host Configuration Protocol (RFC 1531)
2006.03.13

UDP and friends 185 of 220
Internetworking/Internetteknik

- DHCP Message Type - the type of the DHCP message (tag=53)

Message Type	purpose
1	DHCPDISCOVER
2	DHCPOFFER
3	DHCPREQUEST
4	DHCPDECLINE
5	DHCPACK
6	DHCPNAK
7	DHCPRELEASE

- Server Identifier - used in DHCPOFFER and DHCPREQUEST (optionally in DHCPACK and DHCPNAK) messages. Servers include this in the DHCPOFFER to allow the client to distinguish **between** lease offers. DHCP clients indicate which of several lease offers is being accepted by including this in a DHCPREQUEST message. (tag=54)

- Parameter Request List - used by a DHCP client to request values for specified configuration parameters. The client **may** list options in order of preference. The DHCP server **must** try to insert the requested options in the order requested by the client. (tag=55)

Maguire
maguire@it.kth.se

DHCP: Dynamic Host Configuration Protocol (RFC 1531)
2006.03.13

UDP and friends 186 of 220
Internetworking/Internetteknik

- Message - used by a server to provide an error message to client in a DHCPNAK message in the event of a failure. A client may use this in a DHCPDECLINE message to indicate the reason **why** the client declined the offered parameters.(tag=56)

- Maximum DHCP Message Size - specifies the maximum length DHCP message that it is willing to accept. A client may use the maximum DHCP message size option in DHCPDISCOVER or DHCPREQUEST messages, but should not use the option in DHCPDECLINE messages. (tag=57)

- Renewal (T1) Time Value - specifies the time interval from address assignment until the client transitions to the RENEWING state. (tag=58)

- Rebinding (T2) Time Value - specifies the time interval from address assignment until the client transitions to the REBINDING state.(tag=59)

- Class-identifier - used by DHCP clients to optionally identify the type and configuration of a DHCP client. Vendors and sites may choose to define specific class identifiers to convey particular configuration or

Maguire
maguire@it.kth.se

DHCP: Dynamic Host Configuration Protocol (RFC 1531)
2006.03.13

UDP and friends 187 of 220
Internetworking/Internetteknik

other identification information about a client. Servers not equipped to interpret the class-specific information sent by a client **must** ignore it (although it may be reported). (tag=60)

- Client-identifier - used by DHCP clients to specify their unique identifier. DHCP servers use this value to index their database of address bindings. This value is expected to be unique for all clients in an administrative domain. (tag=61)

Maguire
maguire@it.kth.se

DHCP: Dynamic Host Configuration Protocol (RFC 1531)
2006.03.13

UDP and friends 188 of 220
Internetworking/Internetteknik

DHCP's importance

- allows reuse of address, which avoids having to tie up addresses for systems which are not currently connected to the Internet
- avoids user configuration of IP address (avoids mistakes and effort)
- allows recycling of an IP address when devices are scrapped
- …

How big a problem is manual configuration?

A large site (such as DuPont Co. - a large chemical company) has over 65,000 IP addressable devices; or consider what happens if each of the 815,000 Wal-Mart employees has an IP device

Address management software

Product	Vendor	URL
Network Registrar	Cisco	http://www.cisco.com
NetID	Nortel Networks	http://www.nortelnetworks.com
Meta IP 4.1	CheckPoint	http://www.metaip.checkpoint.com
QIP Enterprise 5.0	Lucent Technologies	http://qip.lucent.com

Maguire
maguire@it.kth.se

DHCP's importance
2006.03.13

UDP and friends 189 of 220
Internetworking/Internetteknik

DHCP performance problems

Most implementations of DHCP do a duplicate address detection (DAAD) test after they have picked an address to assign.

An alternative approach to speed up the DHCP process does the duplicate address detection process in the background (in advance) so that you will have a set of recently tested addresses to hand out:

> Jon-Olov Vatn and Gerald Q. Maguire Jr., "The effect of using co-located care-of addresses on macro handover latency", Fourteenth Nordic Tele-traffic Seminar (NTS 14), August 18 - 20, 1998, Lyngby, Denmark.
>
> `http://www.it.kth.se/~vatn/research/nts14-coloc.pdf`

The result is that a DHCP request can be answered in less than 100ms.

Maguire
maguire@it.kth.se

DHCP performance problems
2006.03.13

UDP and friends 190 of 220
Internetworking/Internetteknik

Example of dhcpd.conf

```
### Managed by Linuxconf, you may edit by hand.
### Comments may not be fully preserved by linuxconf.
server-identifier dhcptest1;
default-lease-time 1000;
max-lease-time 2000;
option domain-name            "3ctechnologies.se";
option domain-name-servers  130.237.12.2;
option host-name              "s1.3ctechnologies.se";
option routers                130.237.12.2;
option subnet-mask          255.255.255.0;
subnet 130.237.12.0 netmask 255.255.255.0 {
        range  130.237.12.3 130.237.12.200;
        default-lease-time 1000;
        max-lease-time 2000;
}
subnet 130.237.11.0 netmask 255.255.255.0 {
        range  130.237.11.3 130.237.11.254;
        default-lease-time 1000;
        max-lease-time 2000;
}
```

Maguire
maguire@it.kth.se

Example of dhcpd.conf
2006.03.13

UDP and friends 191 of 220
Internetworking/Internetteknik

DHCP and DNS

- There is no dynamic host name assignment yet.
- Interaction between DHCP and DNS is needed.

For example: once a host is assigned an IP address the DNS should be updated dynamically:

- If the host hasn't got a name: it should assign a name along the IP address assignment (no DNS update is needed).
- If the host has already a name: the DNS should be dynamically updated once the host has gotten a new IP address from DHCP.

The IETF's Dynamic Host Configuration (dhc) Working group

`http://www.ietf.org/html.charters/dhc-charter.html` is working on addressing the issues concerning interaction between DHCP and DNS.

Maguire
maguire@it.kth.se

DHCP and DNS
2006.03.13

UDP and friends 192 of 220
Internetworking/Internetteknik

Trivial File Transfer Procotol (TFTP)

TFTP uses UDP (unlike FTP which uses TCP)

- simple and small

- requires only UDP, IP, and a device driver - easily fits in ROM

- a stop-and-wait protocol

- lost packets detected by timeout and retransmission

- Two operations:
  - Read Request (RRQ)
  - Write Request (RRQ) - for security reasons the file must already exist

- The TFTP server ("tftpd") is generally run setrooted (i.e., it only has access to is own directory) and with a special **user** and **group** ID - since there is no password or other protection of the access to files via TFTP!

- TFTP request is sent to the well know port number (69/udp)

- TFTP server uses an unused ephemeral port for its replies
  - since a TFTP transfer can last for quite some time - it uses another port; thus freeing up the well known port for other requests

Maguire
maguire@it.kth.se

Trivial File Transfer Procotol (TFTP)
2006.03.13

UDP and friends 193 of 220
Internetworking/Internetteknik
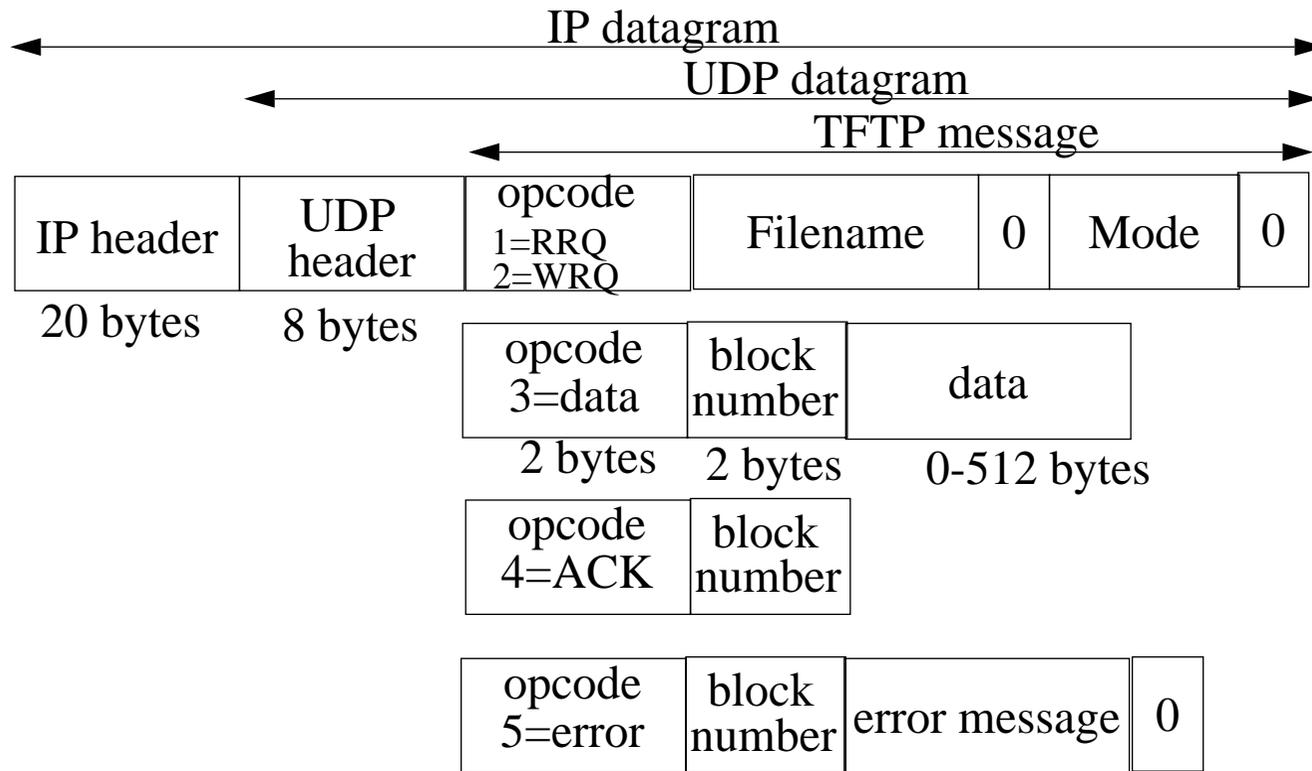
Figure 30: TFTP messages (see Stevens, Vol. 1, figure 15.1, pg. 210)

Filename and Mode ("netascii" or "octet") are both N bytes sequences terminated by a null byte.

Widely used for bootstrapping diskless systems (such as X terminals) and for dumping the configuration of routers (this is where the write request is used)

Maguire
maguire@it.kth.se

Trivial File Transfer Procotol (TFTP)
2006.03.13

UDP and friends 194 of 220
Internetworking/Internetteknik

Mapping names to IP addresses

Host name: **string** **"ccslab1.kth.se"**

Name Resolution: DNS ◄--------

IP address **32 bit** **130.237.15.254**

ARP ↓   RARP ↑   Addr. Resolution

HW address: **48 bit Ethernet MAC address**
"08:00:2B:00:EE:0B"
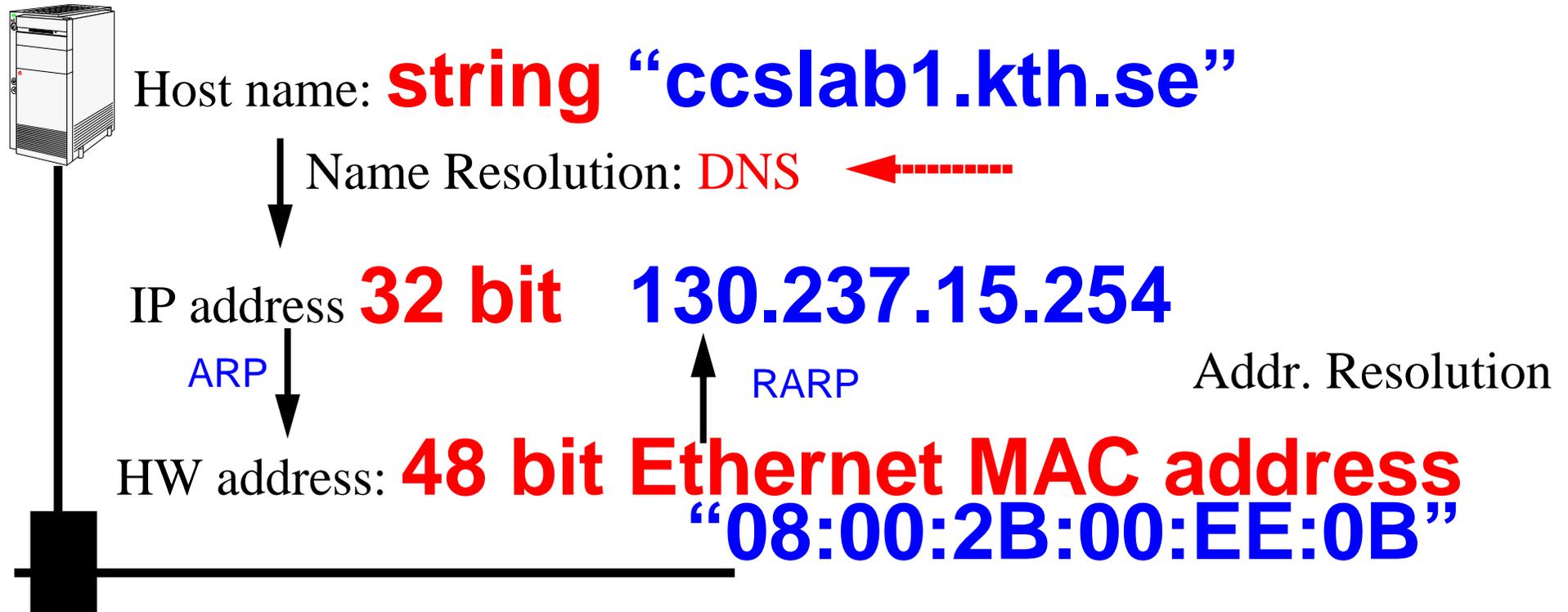
Figure 31: mapping between host names and IP address(es)

Maguire
maguire@it.kth.se

Mapping names to IP addresses
2006.03.13

UDP and friends 195 of 220
Internetworking/Internetteknik

DNS: Domain Name Service (RFC 1034, RFC 1035)

- To make the network more user friendly

- Distributed database (with caching) providing:
  - hostname $\Rightarrow$ IP address, IP address $\Rightarrow$ hostname
  - mailbox $\Rightarrow$ mail server
  - ...

- applications call a "resolver"
  - gethostbyname: hostname $\Rightarrow$ IP address
  - gethostbyaddr: IP address $\Rightarrow$ hostname

- Resolver's contact name servers (see "/etc/resolv.conf")

- DNS names:
  - domain name: list of labels from a root, i.e., www.imit.kth.se
  - Fully Qualified Name (FQDN): a domain name ending in "." - there are no further labels
  - leaves are managed locally through delegation of authority (to a zone) **not** centrally; this allows scaling
  - if a name server does not know the answer it asks other name servers
    - every name server **must** know how to contact a root server

- Uses UDP (for query) and TCP (zone transfer and large record query)

Maguire
maguire@it.kth.se

DNS: Domain Name Service (RFC 1034, RFC 1035)
2006.03.13

UDP and friends 196 of 220
Internetworking/Internetteknik

Zones

A zone is a subtree of the DNS tree which is managed separately.

Each zone must have multiple name servers:

- a primary name server for the zone
  - gets its data from disk files (or other stable store)
  - must know the IP address of one or more root servers

- one or more secondary name servers for the zone
  - get their data by doing a zone transfer from a primary
  - generally query their primary server every ~3 hours

To find a server you may have to walk the tree up to the root or possibly from the root down (but the later is not friendly).

Maguire
maguire@it.kth.se

Zones
2006.03.13

UDP and friends 197 of 220
Internetworking/Internetteknik

DNS Message format

0		16	31
Identification		Parameters	
Number of Questions		Number of Answers	
Numer of authority		Number of Additional	
Question section …			
Answer section …			
Additional Information section …			

Bit or Parameter field	Meaning
0	Operation: 0=Query, 1=Response
1-4	Query type: 0=standard, 1=Inverse
5	Set if answer is authoritative
6	Set if answer is truncate
7	Set if answer is desired
8	Set if answer is available
9-11	reserved
12-15	Response Type: 0=No error, 1=Format error in query, 2=Server failure, 3=Name does not exist

Maguire
maguire@it.kth.se

DNS Message format
2006.03.13

UDP and friends 198 of 220
Internetworking/Internetteknik

Internet's top level domains

(see Stevens, Vol. 1, figure 14.2, pg. 189)

Domain	Description
com	commercial organizations
edu	educational organizations
gov	other U.S. government organizations (see RFC 1811 for policies)
int	international organizations
mil	U.S. Military
net	networks
org	other organizations
arpa	special domain for address to name mappings, e.g., 5.215.237.130.in-addr.arpa
ae	United Arab Emerates
…	
se	Sweden
zw	Zimbabwe

Lots of interest in having subdomains of "com"

♦ companies registering product names, etc. - in some cases asking for 10s of addresses

♦ who gets to use a given name? problems with registered trade marks, who registered the name first, … [How much is a name worth?]

Maguire
maguire@it.kth.se

Internet's top level domains
2006.03.13

UDP and friends 199 of 220
Internetworking/Internetteknik

New top level domains

There is a proposed new set of top level domains and an increase in the number of entities which can assign domain names.

Generic Top Level Domains (gTLDs), November 16, 2000:

.aero	for the entire aviation community
.biz	for business purpose
.coop	for cooperatives
.info	unrestricted
.museum	for museums
.name	for personal names
.pro	for professionals

CORE (Council of Registrars) - operational organization composed of authorized Registrars for managing allocations under gTLDs.

WIPO provides arbitration concerning names:

http://arbiter.wipo.int/domains/gtld/newgtld.html

Maguire
maguire@it.kth.se

New top level domains
2006.03.13

UDP and friends 200 of 220
Internetworking/Internetteknik

Domain registrars

Internet Corporation for Assigned Names and Numbers (ICANN) Accredited Registrars, the full list is at

http://www.icann.org/registrars/accredited-list.html

Even more registrars are on their way to being accredited and operating!

Maguire
maguire@it.kth.se

Domain registrars
2006.03.13

UDP and friends 201 of 220
Internetworking/Internetteknik

Country Code Top-Level Domains (CCTLDs)

http://www.iana.org/cctld/cctld-whois.htm

For Sweden (SE)   SE-DOM, the NIC is: *http://www.nic-se.se/*

	Administrative contact	**Technical contact**
Address	II-Stiftelsen Sehlstedtsgatan 7 SE-115 28 Stockholm Sweden	Network Information Centre Sweden NIC-SE Box 5774 SE-114 87 Stockholm Sweden
e-mail	ii-stiftelsen@iis.se	hostmaster@nic-se.se
phone	+46 8 56849050	+46 8 54585700
fax	+46 8 50618470	+46 8 54585729

The above is from *http://www.iana.org/root-whois/se.htm*

URL for registration services: *http://www.iis.se/*

Maguire
maguire@it.kth.se
Country Code Top-Level Domains (CCTLDs)
2006.03.13
UDP and friends 202 of 220
Internetworking/Internetteknik

Resource Records (RR)

See Stevens, Vol. 1, figure 14.2, pg. 201 (augmented with additional entires)

Record type	Description
A	an IP address. Defined in RFC 1035
AAAA	an IPv6 address. Defined in RFC 1886
PTR	pointer record in the in-addr.arpa format. Defined in RFC 1035.
CNAME	canonical name≡ alias (in the format of a domain name). Defined in RFC 1035.
HINFO	Host information. Defined in RFC 1035.
MX	Mail eXchange record. Defined in RFC 1035.
NS	authoritative Name Server (gives authoritative name server for this domain).Defined in RFC 1035.
TXT	other attributes. Defined in RFC 1035.
AFSDB	AFS Data Base location. Defined in RFC 1183.
ISDN	ISDN. Defined in RFC 1183.
KEY	Public key. Defined in RFC 2065.
KX	Key Exchanger. Defined in RFC 2230.
LOC	Location. Defined in RFC 1876.
MG	mail group member. Defined in RFC 1035.
MINFO	mailbox or mail list information. Defined in RFC 1035.
MR	mail rename domain name. Defined in RFC 1035.
NULL	null RR. Defined in RFC 1035.
NS	Name Server. Defined in RFC 1035.

Maguire
maguire@it.kth.se

Resource Records (RR)
2006.03.13

UDP and friends 203 of 220
Internetworking/Internetteknik

See Stevens, Vol. 1, figure 14.2, pg. 201 (augmented with additional entires)

Record type	Description
NSAP	Network service access point address. Defined in RFC 1348. Redefined in RFC 1637. Redefined in RFC 1706.
NXT	Next. Defined in RFC 2065.
PX	Pointer to X.400/RFC822 information. Defined in RFC 1664.
RP	Responsible Person. Defined in RFC 1183.
RT	Route Through. Defined in RFC 1183.
SIG	Cryptographic signature. Defined in RFC 2065.
SOA	Start Of Authority. Defined in RFC 1035.
SRV	Server. DNS Server resource record -- RFC 2052, for use with DDNS.
TXT	Text. Defined in RFC 1035.
WKS	Well-Known Service. Defined in RFC 1035.
X25	X25. Defined in RFC 1183.

Note that an number of the RR types above are for experimental use.

Name of an organization:

ISI.EDU.   PTR   0.0.9.128.IN-ADDR.ARPA.

Maguire
maguire@it.kth.se

Resource Records (RR)
2006.03.13

UDP and friends 204 of 220
Internetworking/Internetteknik

Network names

Conventions:

- it.kth.se includes all the computers in the KTH/SU IT-University
- kth.se includes all the computers at KTH
- …

As resource records:

> set querytype=any

> kth.se

…

Non-authoritative answer:

kth.se   internet address = 130.237.72.201

kth.se

origin = kth.se

mail addr = hostmaster.kth.se

serial = 2002011500

refresh = 3600 (1H)

retry   = 600 (10M)

expire = 604800 (1W)

minimum ttl = 86400 (1D)

kth.se	nameserver = kth.se
kth.se	nameserver = nic.lth.se
kth.se	nameserver = ns.kth.se

Authoritative answers can be found from:

kth.se	nameserver = kth.se
kth.se	nameserver = nic.lth.se
kth.se	nameserver = ns.kth.se
kth.se	internet address = 130.237.x.y
nic.lth.se	internet address = 130.235.z.w
ns.kth.se	internet address = 130.237.m.n

ARPANET.ARPA.	PTR	0.0.0.10.IN-ADDR.ARPA.
isi-net.isi.edu.	PTR	0.0.9.128.IN-ADDR.ARPA.

Maguire
maguire@it.kth.se

Network names
2006.03.13

UDP and friends 206 of 220
Internetworking/Internetteknik

Example:

```
$ORIGIN it.kth.se.
@                          1D IN SOA        bbbb hostmaster (
                                            2002012001        ; serial
                                            8H                ; refresh
                                            2H                ; retry
                                            2W                ; expiry
                                            8H )              ; minimum

          1D      IN NS        ns.ele.kth.se.
          1D      IN NS        ns.kth.se.
          1D      IN MX        0 mail
          1D      IN A         130.237.x.y
          1D      IN AFSDB     1 xxxx
          1D      IN AFSDB     1 yyyy
          1D      IN AFSDB     1 zzzz
```

Maguire
maguire@it.kth.se

Example:
2006.03.13

UDP and friends 207 of 220
Internetworking/Internetteknik

MX information

```
> set querytype=MX
> kth.se

…
kth.se                          preference = 0, mail exchanger = mail1.kth.se
kth.se                          nameserver = kth.se
kth.se                          nameserver = nic.lth.se
kth.se                          nameserver = ns.kth.se
mail1.kth.se                    internet address = 130.237.32.62
kth.se                          internet address = 130.237.72.201
nic.lth.se                      internet address = 130.235.20.3
ns.kth.se                       internet address = 130.237.72.200
```

Another examine in MX RR format:

```
1D      IN MX       10 xxx.e.kth.se.
1D      IN MX       20 yyy.e.kth.se.
```

Host names and info

How to give your host a name?

see RFC 1178: Choosing a Name for Your Computer

Internet Addresses: A second address for your host?

- to have multiple addresses for you computer, see section on "ifconfig"

Hostinfo (HINFO)

> set querytype=HINFO

> kth.se

…

kth.se   CPU = sun-4/60   OS = unix

Entry	owner	clas	TTL	RR type	value	comment
xxxx			1D	IN HINFO	"PC" "FLINUX"	; "CPU" "OS"
			1D	IN A	130.237.x.y	

Maguire
maguire@it.kth.se

Host names and info
2006.03.13

UDP and friends 209 of 220
Internetworking/Internetteknik

Storing other attributes - TXT records

The general syntax is:

<owner> <class> <ttl> TXT "<attribute name>=<attribute value>"

Examples:

host.widgets.com   IN   TXT   "printer=lpr5"

sam.widgets.com   IN   TXT   "favorite drink=orange juice"

For more information see:

RFC 1464: *Using the Domain Name System To Store Arbitrary String Attributes*

Maguire
maguire@it.kth.se

Storing other attributes - TXT records
2006.03.13

UDP and friends 210 of 220
Internetworking/Internetteknik

Configuring DNS

- ## Configuring the BIND resolver
  - /etc/resolv.config

- ## Configuring the BIND nameserver (named)
  - /etc/named.boot   or /etc/named.config

- ## Configuring the nameserver database files (zone files)
  - named.hosts   the zone file that maps hostnames to IP addresses
  - named.rev   the zone file that maps IP addresses to hostnames

Maguire
maguire@it.kth.se

Configuring DNS
2006.03.13

UDP and friends 211 of 220
Internetworking/Internetteknik

Root servers

ROOT-SERVERS.NET		IP address(es)	Home ASN	Location(s)
A	Verisign Global Registry Services	198.41.0.4	19836	Herndon, VA, US
B	EP.Net	192.228.79.201 2001:478:65::53		Marina del Rey, CA, US
C	Cogent Communications	192.33.4.12	2149	Herndon VA; Los Angeles; New York City; Chicago
D	University of Maryland	128.8.10.90	27	College Park, MD, US
E	NASA Ames Research Centre	192.203.230.10	297	Mountain View, CA, US
F	Internet Systems Consortium (ISC)	192.5.5.241 2001:500::1035	3557	Ottawa; Palo Alto; San Jose CA;New York City; San Francisco;Madrid; Hong Kong; Los Angeles;Rome; Auckland; Sao Paulo;Beijing; Seoul; Moscow; Taipei;Dubai; Paris; Singapore; Brisbane;Toronto; Monterrey; Lisbon;Johannesburg; Tel Aviv; Jakarta;Munich; Osaka; Prague
G	US Department of Defence	192.112.36.4	568	Vienna, VA, US
H	US Army Research Lab	128.63.2.53 2001:500:1::803f:235	13	Aberdeen, MD, US
I	Autonomica/NORDUnet	192.36.148.17	29216	Stockholm; Helsinki; Milan;London; Geneva; Amsterdam;Oslo; Bangkok; Hong Kong;Brussels; Frankfurt; Ankara; Bucharest;Chicago; Washington DC;Tokyo; Kuala Lumpur;Palo Alto; Wellington
J	Verisign Global Registry Services	192.58.128.30	26415	Herndon, VA, US
K	Reseaux IP Europeens (RIPE)	193.0.14.129 2001:7fd::1	25152	London (UK); Amsterdam (NL);Frankfurt (DE); Athens (GR);Doha (QA); Milan (IT);Reykjavik (IS); Helsinki (FI);Geneva (CH); Poznan (PL);Budapest (HU)
L	IANA	198.32.64.12	20144	Los Angeles, CA, US
M	WIDE Project	202.12.27.33 2001:dc3::35	7500	Tokyo; Seoul (KR); Paris (FR)

see *http://www.root-servers.org/*

Maguire
maguire@it.kth.se

Root servers
2006.03.13

UDP and friends 212 of 220
Internetworking/Internetteknik

Load leveling [22]

For example, f.root-servers.net has a **single** IP address (192.5.5.241), but requests sent to 192.5.5.241 are routed to *different* nameservers

This routing can depend on:

- where the request is made from
- what the load on each of these names servers is

Note: this is transparent to the host which sent the request to F

ISC uses Hierarchical Anycast routing to do this, with some of the servers being:

- large, redundant, … installations serving the *global* internet
- small installations serving a *local* region
- 28 nodes as of Feb. 2005

Maguire
maguire@it.kth.se

Load leveling [22]
2006.03.13

UDP and friends 213 of 220
Internetworking/Internetteknik

F root nameserver nodes

	Site Code	Location	IPv4/IPv6	Node Type
Asia Pacific	AKL1	Auckland	IPv4	Local Node
	BNE1	Brisbane	IPv4	Local Node
	CGK1	Jakarta	IPv4	Local Node
	HKG1	Hong Kong	IPv4	Local Node
	KIX1	Osaka	IPv4 and IPv6	Local Node
	PEK1	Beijing	IPv4	Local Node
	SEL1	Seoul	IPv4 and IPv6	Local Node
	TPE1	Taipei	IPv4	Local Node
Americas	AKL1	São Paulo	IPv4	Local Node
	LAX1	Los Angeles	IPv4 and IPv6	Local Node
	LGA1	New York	IPv4 and IPv6	Local Node
	MTY1	Monterrey	IPv4	Local Node
	PAO1	Palo Alto	IPv4 and IPv6	Global Node
	SFO2	San Francisco	IPv4 and IPv6	Global Node
	SQL1	Redwood City	IPv4 and IPv6	Local Node
	SJC1	San Jose	IPv4	Local Node
	YOW1	Ottawa	IPv4 and IPv6	Local Node
	YYZ1	Toronto	IPv4	Local Node
Rest of World	CDG1	Paris	IPv4 and IPv6	Local Node
	DXB1	Dubai	IPv4	Local Node
	JNB1	Johannesburg	IPv4	Local Node
	LIS1	Lisbon	IPv4 and IPv6	Local Node
	MAD1	Madrid	IPv4	Local Node
	MUC1	Munich	IPv4 and IPv6	Local Node
	PRG1	Prague	IPv4	Local Node
	ROM1	Rome	IPv4	Local Node
	SVO1	Moscow	IPv4	Local Node
	TLV1	Tel Aviv	IPv4	Local Node

Maguire
maguire@it.kth.se

F root nameserver nodes
2006.03.13

UDP and friends 214 of 220
Internetworking/Internetteknik

Where is f.root-servers.net ?

traceroute to f.root-servers.net (192.5.5.241), 30 hops max, 40 byte packets

1	net212a.imit.kth.se (130.237.212.2)	1 ms	1 ms	0 ms
2	cn4-kf4-p2p.gw.kth.se (130.237.211.205)	1 ms	1 ms	1 ms
3	ea4-cn4-p2p.gw.kth.se (130.237.211.102)	1 ms	1 ms	1 ms
4	kth2-ea4-b.gw.kth.se (130.237.211.241)	1 ms	1 ms	1 ms
5	stockholm4-SRP2.sunet.se (130.242.85.66)	1 ms	1 ms	1 ms
6	130.242.82.49 (130.242.82.49)	1 ms	1 ms	1 ms
7	se-kth.nordu.net (193.10.252.177)	1 ms	2 ms	4 ms
8	so-1-0.hsa2.Stockholm1.Level3.net (213.242.69.21)	4 ms	5 ms	4 ms
9	so-4-1-0.mp2.Stockholm1.Level3.net (213.242.68.205)	14 ms	8 ms	8 ms
10	as-1-0.bbr2.London1.Level3.net (212.187.128.25)	36 ms	36 ms	36 ms
11	as-0-0.bbr1.NewYork1.Level3.net (4.68.128.106)	106 ms	104 ms	104 ms
12	ae-0-0.bbr2.SanJose1.Level3.net (64.159.1.130)	180 ms	247 ms	180 ms
13	so-14-0.hsa4.SanJose1.Level3.net (4.68.114.158)	*	178 ms	178 ms
14	ISC-Level3-fe.SanJose1.Level3.net (209.245.146.219)	178 ms	180 ms	178 ms
15	f.root-servers.net (192.5.5.241)	180 ms	178 ms	180 ms

Maguire
maguire@it.kth.se

Where is f.root-servers.net ?
2006.03.13

UDP and friends 215 of 220
Internetworking/Internetteknik

Where is i.root-servers.net ?

traceroute to i.root-servers.net (192.36.148.17), 30 hops max, 40 byte packets

1	net212a.imit.kth.se (130.237.212.2)	1 ms	0 ms	0 ms
2	cn4-kf4-p2p.gw.kth.se (130.237.211.205)	1 ms	1 ms	4 ms
3	cn5-cn4-p2p.gw.kth.se (130.237.211.201)	4 ms	4 ms	4 ms
4	kth1-cn5-p2p.gw.kth.se (130.237.211.41)	4 ms	4 ms	4 ms
5	stockholm3-SRP2.sunet.se (130.242.85.65)	4 ms	4 ms	4 ms
6	ge-2-2.cyb-gw.sth.netnod.se (194.68.123.73)	2 ms	1 ms	1 ms
7	ge-2-1.icyb-gw.sth.netnod.se (192.36.144.190)	1 ms	1 ms	2 ms
8	srp-1-1.ibyb-gw.sth.netnod.se (192.36.144.235)	1 ms	1 ms	1 ms
9	ge-0-0.r1.sth.dnsnode.net (194.146.105.187)	1 ms	2 ms	2 ms
10	i.root-servers.net (192.36.148.17)	3 ms	3 ms	2 ms

Maguire
maguire@it.kth.se

Where is i.root-servers.net ?
2006.03.13

UDP and friends 216 of 220
Internetworking/Internetteknik

Dynamic Domain Name System (DDNS)

```
                        ┌──────────────┐
                        │    Root      │
                        │   Server     │
                        └──────────────┘
          ┌────────────┬──────┴────────┬──────────────────┐
          ▼            ▼               ▼                   ▼
   ┌────────────┐ ┌────────────┐ ┌────────────┐      ┌────────────┐
   │ Server for │ │ Server for │ │ Server for │ ...  │ Server for │
   │    .se     │ │   .com     │ │   .edu     │      │    .xx     │
   └────────────┘ └────────────┘ └────────────┘      └────────────┘
          │
          ▼
   ┌────────────┐
   │ Server for │
   │   kth.se   │
   └────────────┘
```
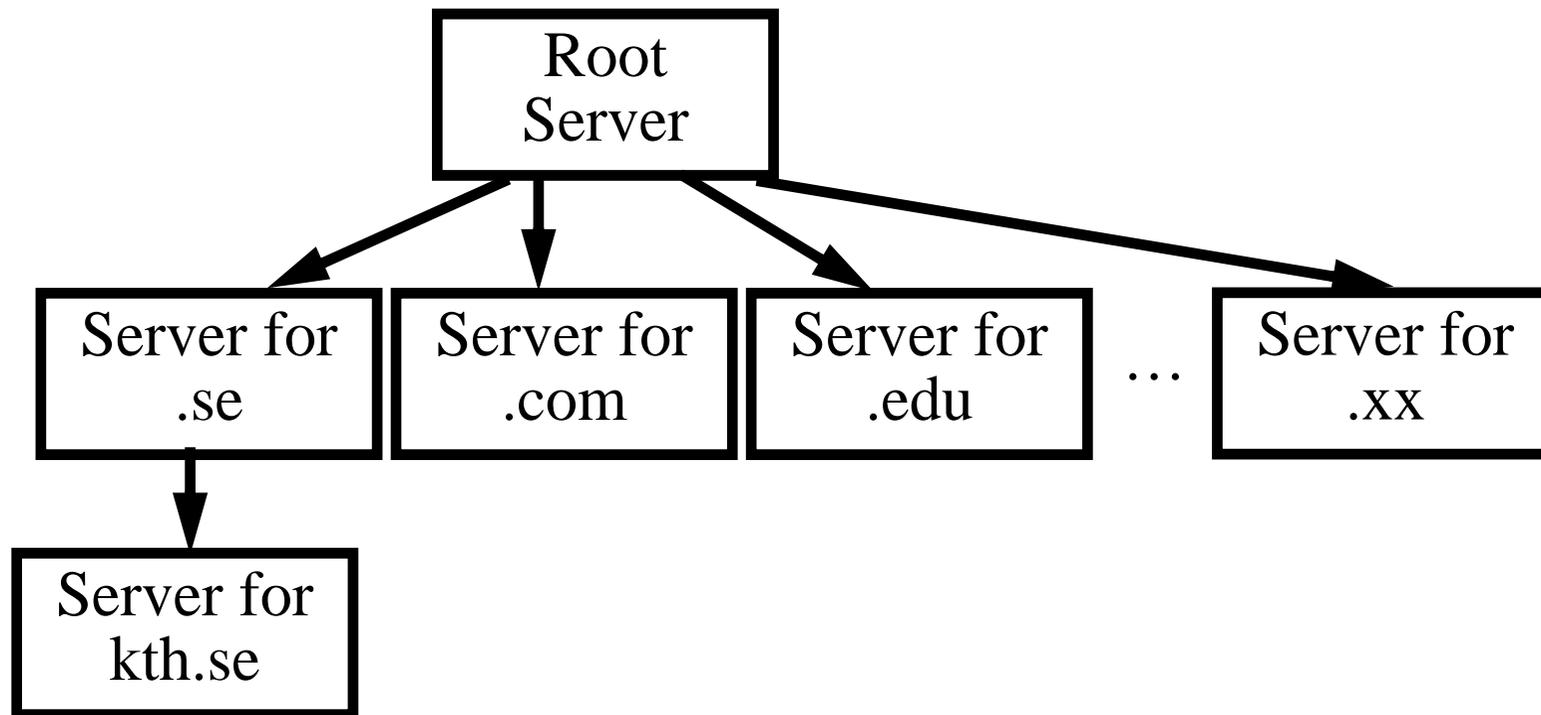
Host Name	IP-address
host_a	130.237.x.1
host_b	130.237.x.2
host_c	130.237.x.3
host_d	130.237.x.4
mobile1	c/o address <<< we can update this dynamically

Maguire
maguire@it.kth.se

Dynamic Domain Name System (DDNS)
2006.03.13

UDP and friends 217 of 220
Internetworking/Internetteknik

DDNS

RFC 2136: Dynamic Updates in the Domain Name System (DNS UPDATE)

- add or delete resource records

RFC 2052: A DNS RR for specifying the location of services (DNS SRV)

- When a SRV-cognizant web-browser wants to retrieve

  http://www.asdf.com/

it does a lookup of

  http.tcp.www.asdf.com

RFC 2535: Domain Name System Security Extensions (DNSSec)

Maguire
maguire@it.kth.se

DDNS
2006.03.13

UDP and friends 218 of 220
Internetworking/Internetteknik

Summary

This lecture we have discussed:

- UDP
- BOOTP
- DHCP
- DNS, DDNS

Maguire
maguire@it.kth.se

Summary
2006.03.13

UDP and friends 219 of 220
Internetworking/Internetteknik

References

[22] Joe Abley, f.root-servers.net, NZNOG 2005, February 2005, Hamilton, NZ

   http://www.isc.org/pubs/pres/NZNOG/2005/F%20Root%20Server.pdf

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 5: TCP, HTTP, RPC, NFS, X

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapter 12

KTH Information and Communication Technology

Lecture 4: Outline

- TCP
- HTTP
- Web enabled devices
- RPC, XDR, and NFS
- X Window System, and
- some tools for looking at these protocols

Maguire
maguire@it.kth.se

Lecture 4: Outline
2006.03.13

TCP, HTTP, RPC, NFS, X 222 of 297
Internetworking/Internetteknik

Transport layer protocols

- User Datagram Protcol (UDP)
  - Connectionless unreliable service
- Transmission Control Protocol (TCP) <<< today's topic
  - Connection-oriented reliable stream service
- Stream Control Transmission Protocol (SCTP)
  - a modern transmission protocol with many facilities which the user can chose from

Maguire
maguire@it.kth.se

Transport layer protocols
2006.03.13

TCP, HTTP, RPC, NFS, X 223 of 297
Internetworking/Internetteknik

Transmission Control Protocol (TCP)

TCP provides a connection oriented, reliable, byte stream service[23].

- ## TCP utilizes full-duplex connections
  - Note: There are just **two** endpoints
- ## TCP applications write 8-bit bytes to a stream and read bytes from a stream
  - TCP decides how much data to send (not the application) - each unit is a **segment**
  - There are no records (or record makers) - just a stream of bytes $\Rightarrow$ the receiver can't tell how much the sender wrote into the stream at any given time
- ## TCP provides reliability
  - Acknowledgements, timeouts, retransmission, …
- ## TCP provides flow control
- ## TCP trys to avoid causing congestion

Maguire
maguire@it.kth.se

Transmission Control Protocol (TCP)
2006.03.13

TCP, HTTP, RPC, NFS, X 224 of 297
Internetworking/Internetteknik

Applications which use TCP

Lots of applications have been implemented on top of the TCP, such as:

- TELNET — provides a virtual terminal {emulation}
- FTP — used for file transfers
- SMTP — forwarding e-mail
- HTTP — transport data in the World Wide Web

Here we will focus on some features not covered in the courses: Telesys, gk and Data and computer communication.

Maguire
maguire@it.kth.se

Transmission Control Protocol (TCP)
2006.03.13

TCP, HTTP, RPC, NFS, X 225 of 297
Internetworking/Internetteknik

TCP Header

0	7 8	15 16	23 24	31

16 bit source port	16 bit destination port

32 bit sequence number

32 bit acknowledgement number

4 bit header length	reserved (6 bits)	U R G	A C K	P S H	R S T	S Y N	F I N	16 bit window size

16 bit TCP checksum	16 bit urgent pointer

options (if any)

data (if any - padded to a multiple of 16 bits)

(20 bytes)

Figure 32: IP header (see Stevens, Vol. 1. inside cover or Forouzan figure 12.5 pg. 282)

Just as with UDP, TCP provides de/multiplexing via the 16 bit source and destination ports.

The 4 bit header length indicates the number of **4-byte words** in the TCP header

Maguire
maguire@it.kth.se

TCP Header
2006.03.13

TCP, HTTP, RPC, NFS, X 226 of 297
Internetworking/Internetteknik

TCP header continued

Reliability is provided by the 32 bit sequence number which indicates the byte offset in a stream of the first byte in this segment and a 32 bit acknowledgement number which indicates the next byte which is **expected**.

- The initial sequence number (ISN) is a random 32 bit number.
- Note that the acknowledgement is **piggybacked** in each TCP segment
  - TCP maintains a timer for *each* segment. If an acknowledgement is not received before the timeout, then TCP retransmits the segment
  - When TCP receives data it sends an acknowledgement back to sender
- TCP applies an end-to-end checksum on its header and data
  - The checksum is mandatory - but otherwise similar to the UDP checksum
- TCP resequences data at the receiving side ⇒ all the bytes are delivered in order to the receiving application
- TCP discards duplicate data at the receiving side

Urgent pointer - specifies that the stream data is offset and that the data field begins with "urgent data" which is to bypass the normal stream - for example ^C

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 227 of 297
Internetworking/Internetteknik

Control field - indicates the purpose & contents of this segment:

Flag	Description
URG	The urgent pointer is valid
ACK	The acknowledgement number is valid
PSH	Push the data, i.e., the receiver should immediately pass all the data to the application $\Rightarrow$ emptying the receiver's buffer
RST	Connection must be rest
SYN	Synchronize the sequence numbers
FIN	Terminate the connection (from the **sender's** point of view)

We will see how these bits are used as we examine each of them later.

The window size (or more exactly the receive window size (rwnd)) - indicates how many bytes the receiver is prepared to receive (this number is relative to the acknowledgement number).

Options - as with UDP there can be up to 40 bytes of options (we will cover these later)

Maguire
maguire@it.kth.se
TCP header continued
2006.03.13
TCP, HTTP, RPC, NFS, X 228 of 297
Internetworking/Internetteknik

Connection establishment

client   **server**

3-way handshake:

- Guarantees both sides are ready to transfer data
- Allows both sides to agree on initial sequence numbers

time

SYN, seq=x

SYN, seq=y, ACK=x+1

ACK=y+1
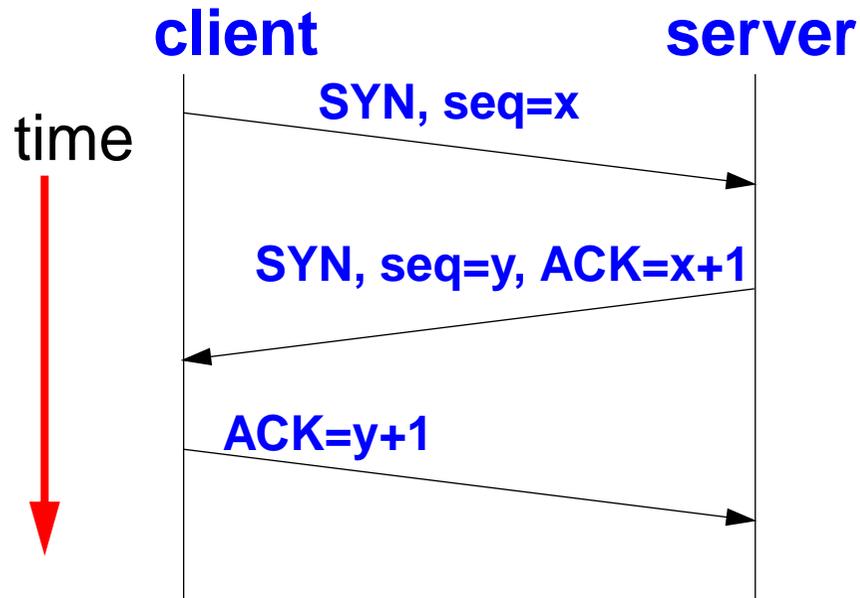
See Forouzan figure 12.9 page 286

Initial sequence number (ISN) **must** be chosen so that each instance of a specific TCP connection between two end-points has a different ISN.

The entity initiating the connection is (normally) called the "client".

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 229 of 297
Internetworking/Internetteknik

SYN Flooding Attack

It is clear that if a malacioous user simply sends a lot of SYN sgements to a target machine (with faked source IP addresses) $\Rightarrow$ this machine will spend a lot of resources to set up TCP connections which subsequently never occur.

As the number of TCP control blocks and other resources are finite

- legitimate connection requests can't be answered
- the target machine might even crash

The result is to deny service, this is one of many Denial of Services (DoS) Attacks

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 230 of 297
Internetworking/Internetteknik

Connection teardown
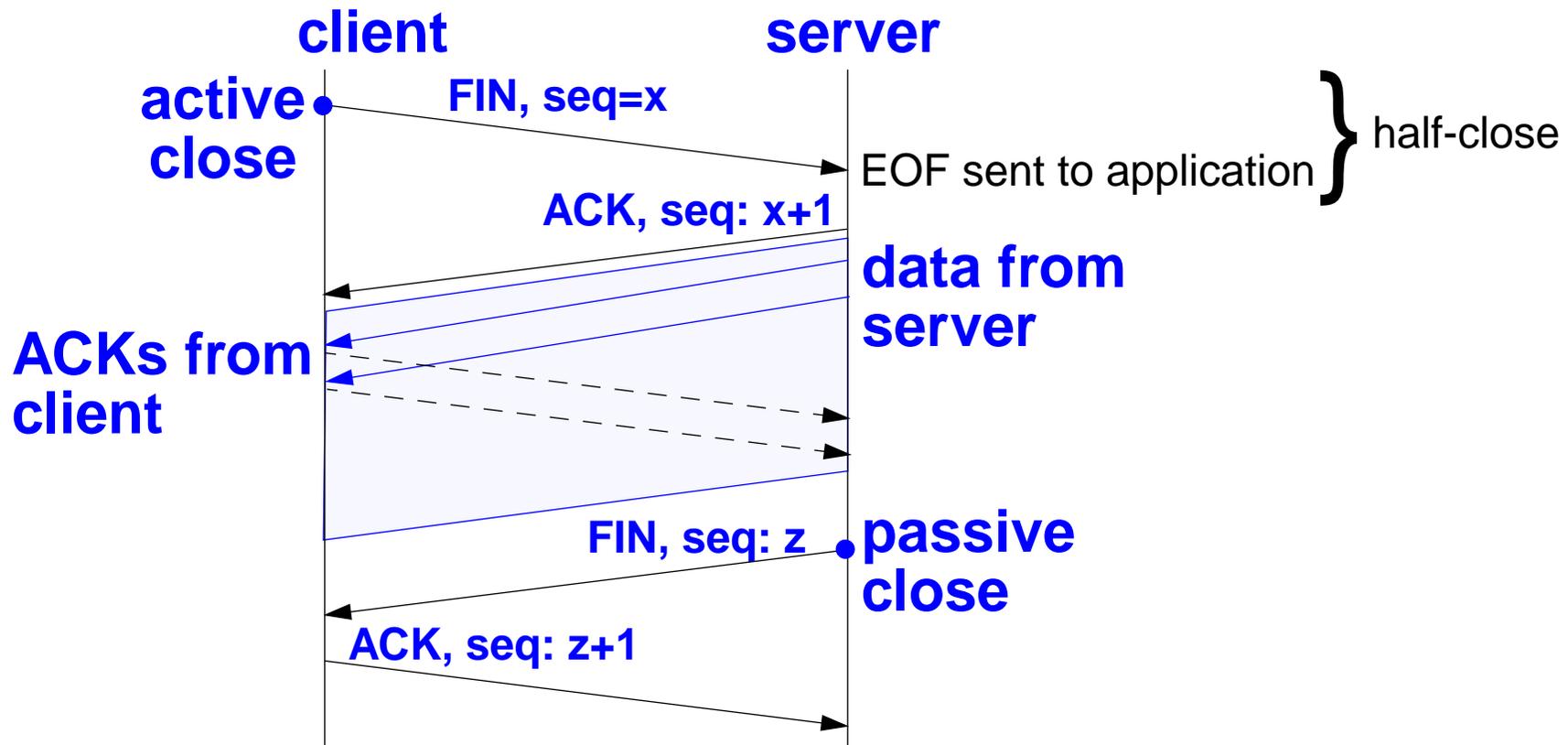
See Forouzan figure 12.12 page 291

- **Note:** it takes 4 segments to complete the close.
- Normally, the client performs an **active close** and the server performs **passive close.**

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 231 of 297
Internetworking/Internetteknik

TCP options

These options are used to convey additional information to the destination or to align another options

- ## Single-byte Options
  - No operation
  - End of option

- ## Multiple-byte Options
  - Maximum segment size
  - Window scale factor
  - Timestamp

A TCP header can have up to 40 bytes of optional information.

Maximum Segment Size

- The Maximum Segment Size (MSS) is the largest amount of data TCP will send to the other side

- MSS can be announced in the options field of the TCP header during connection establishment

- If a MSS is not announced $\Rightarrow$ a default value of 536 is assumed

- In general, the larger MSS is the better -- until fragmentation occurs
  - As when fragmentation occurs the overhead increases!

Maguire

maguire@it.kth.se

TCP header continued

2006.03.13

TCP, HTTP, RPC, NFS, X 233 of 297

Internetworking/Internetteknik

Sliding window Flow control

- receiver: **offered window** - acknowledges data sent and what it is prepared to receive
  - thus the sender can send an ACK, but with a offered window of 0
  - later the sender sends a **window update** with a non-zero offered window size
  - the receiver can increase or decrease this window size as it wants

- sender: **usable window** - how much data it is prepared to send immediately

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 234 of 297
Internetworking/Internetteknik
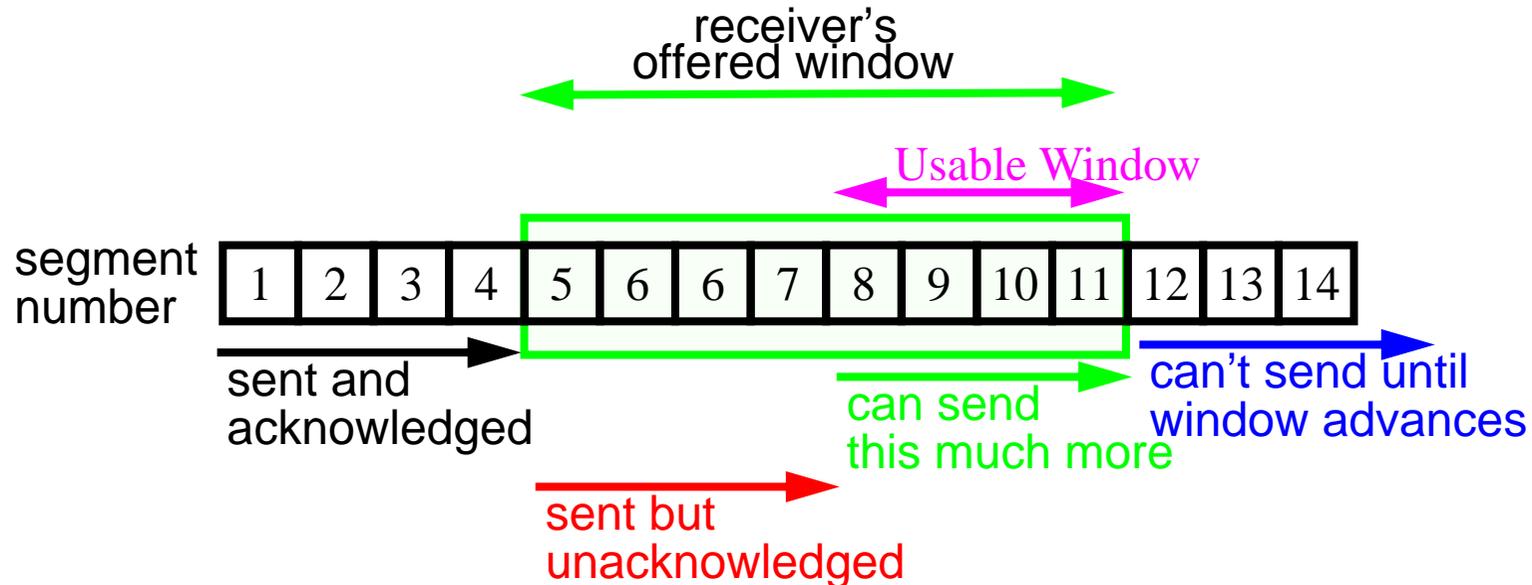
Window size

Increasing window size can improve performance - more recent systems have increased buffer size ranging from 4096 ... 16,384 bytes. The later produces ~40% increase in file transfer performance on an ethernet.

Socket API allows user to change the size of the send and receive buffers.

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 235 of 297
Internetworking/Internetteknik

Flow for an rlogin session

client   **server**

keystroke → seq=x; one data byte

ACK, seq: x+1

**display
the character**   ←   seq=y; one data byte   **echo of keystroke
sent by server**

ACK, seq=y+1
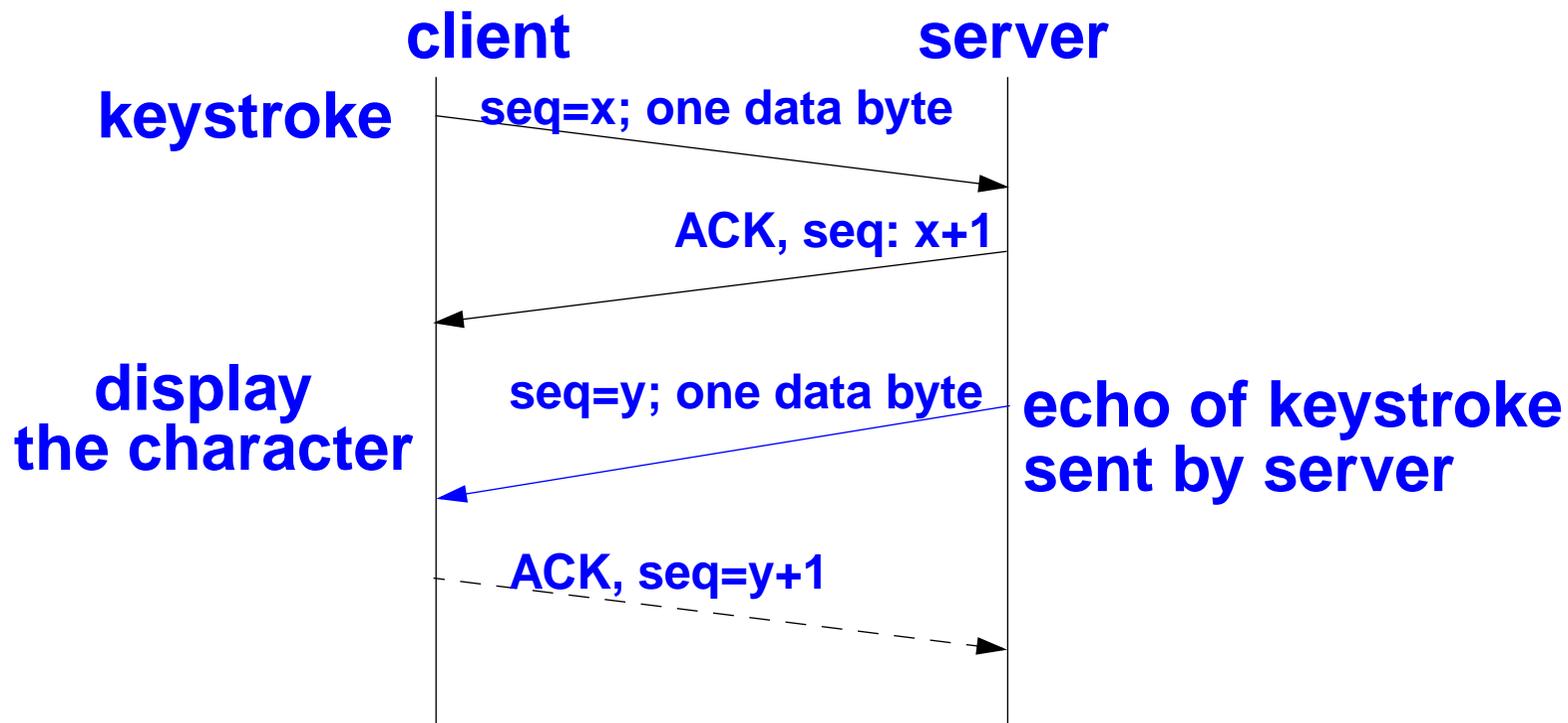
See Forouzan figure 12.12 page 291

Thus **each** keystroke not only generates a byte of data for the remote application which has to be sent in a segments, but this will trigger an ACK along with an echo & its ACK $\Rightarrow$ generates 3 more segments! All to send a single byte of user data!!!

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 236 of 297
Internetworking/Internetteknik

Silly Window Syndrome

If receiver advertises a small window, then sender will send a small amount of data, which fills receivers window, … .

To prevent this behavior:

- sender does not transmit unless:
  - full-size segment can be sent OR
  - it can send at least 1/2 maximum sized window that the other has ever advertised
  - we can send everything we have and are not expecting an ACK or Nagle algorithm is disabled

- receiver must not advertise small segments
  - advertise **min**(a full-size segment, 1/2 the receive buffers space)
  - delayed ACKs

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 237 of 297
Internetworking/Internetteknik

Nagle Algorithm

telnet/rlogin/... generate a packet (41 bytes) for each 1 byte of user data

- these small packets are called "tinygrams"
- not a problem on LANs
- adds to the congestion on WANs

Nagle Algorithm

- each TCP connection can have only one outstanding (i.e., unacknowledged) small segment (i.e., a tinygram)
- while waiting - additional data is accumulated and sent as one segment when the ACK arrives
- self-clock: the faster ACKs come, the more often data is sent
  - thus automatically on slow WANs fewer segments are sent

Round trip time on a typical ethernet is ~16ms (for a single byte to be sent, acknowledged, and echoed) - thus to generate data faster than this would require typing faster than 60 characters per second! Thus rarely will Nagle be invoked on a LAN.

Maguire
maguire@it.kth.se
TCP header continued
2006.03.13
TCP, HTTP, RPC, NFS, X 238 of 297
Internetworking/Internetteknik

Disabling the Nagle Algorithm

But sometimes we need to send a small message - without waiting (for example, handling a mouse event in the X Window System) - therefore we set:

- `TCP_NODELAY` on the socket
- Host Requirements RFC says that hosts **must** implement the Nagle algorithm, but there **must** be a way to disable it on individual connections.

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 239 of 297
Internetworking/Internetteknik

Delayed acknowledgements

Rather than sending an ACK immediately, TCP waits ~200ms hoping that there will be data in the reverse direction - thus enabling a piggybacked ACK.

- Host Requirements RFC states the delays must be less than 500ms
- Implementations often use a periodic 200ms timer - rather than setting a timer specifically for computing this delay
  - similar to the periodic 500ms timer used for detecting timeouts

A complex protocol with several timers leads to complex behavior!

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 240 of 297
Internetworking/Internetteknik

Resulting bulk data flow

Every segment cares a full MSS worth of data.

Typically an ACK every other segment.

Maguire
maguire@it.kth.se
TCP header continued
2006.03.13
TCP, HTTP, RPC, NFS, X 241 of 297
Internetworking/Internetteknik
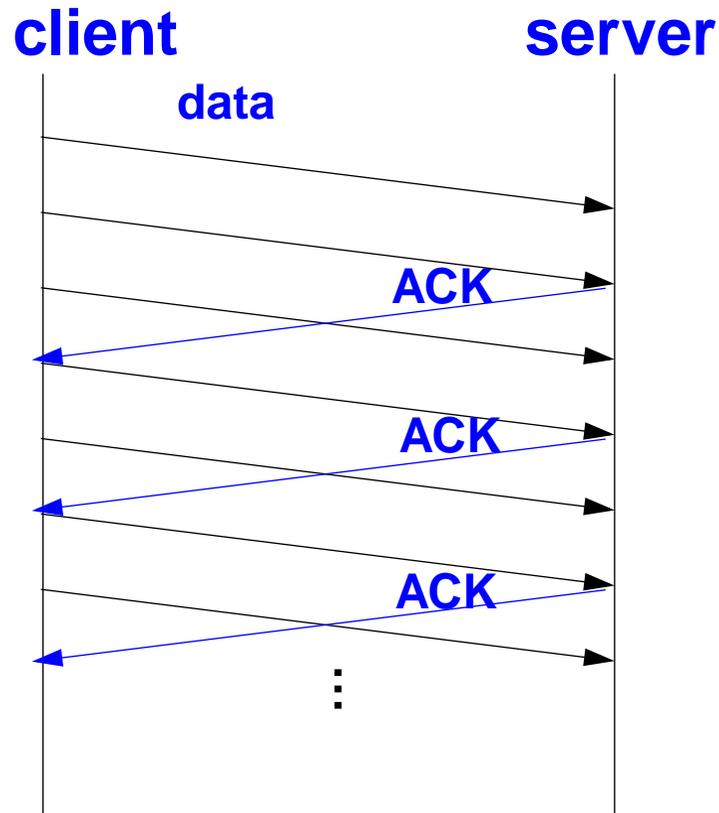
Bandwidth-Delay Product

How large should the window be for optimal throughput?

Calculate the capacity of the pipe as:

- capacity(bits) = bandwidth(bits/sec) * RTT(sec)
- This is the size the receiver advertised window should have for optimal throughput.

Example:

T1 connection across the US:
```
capacity = 1.544Mbit/s * 60ms = 11,580 bytes
```
Gigabit connection across the US:
```
capacity = 1Gbit/s * 60ms = 7,500,000 bytes!
```

However, the window size field is only 16 bits $\Rightarrow$ maximum value of 65535 bytes

For Long Fat Pipes we can use the window scale option to allow much larger window sizes.

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 242 of 297
Internetworking/Internetteknik

Congestion Avoidance

So far we have assumed that the sender is only limited by the receiver's available buffer space. But if we inject lots of segments into a network - upto window size advertised by receiver

- works well if the hosts are on the same LAN
- if there are routers (i.e., queues) between them and if the traffic arrives faster than it can be forwarded, then either the packets have to be
    - buffered or
    - thrown away - we refer to this condition as congestion

Lost packets lead to retransmission by the sender

This adds even more packets to the network $\Rightarrow$ **network collapse**

Therefore we need to be able to reduce the window size to avoid congestion.

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 243 of 297
Internetworking/Internetteknik

Congestion Control

We introduce a Congestion Window

- Thus the sender's window size will be determined both by the receiver and in reaction to congestion in the network

Sender maintains 2 window sizes:

- Receiver-advertised window (rwnd)
  - advertised window is flow control imposed by **receiver**
- Congestion window (CWND)
  - congestion window is flow control imposed by **sender**

Actual window size = min(rwnd, CWND)

To deal with congestion, sender uses several strategies:

- Slow start
- Additive increase of CWND
- Multiplicative decrease of CWND

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 244 of 297
Internetworking/Internetteknik

Slow start

In 1989, Van Jacobson introduced **slow start** based on his analysis of actual traffic and the application of control theory. All TCP implementations are now required to support slow start.

- the rate at which new packets should be injected into the network is the rate at which acknowledgements are returned

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 245 of 297
Internetworking/Internetteknik

- cwnd starts at number of bytes in one segment (as announced by other end) and increases exponentially with successfully received **cwnd** worth of data
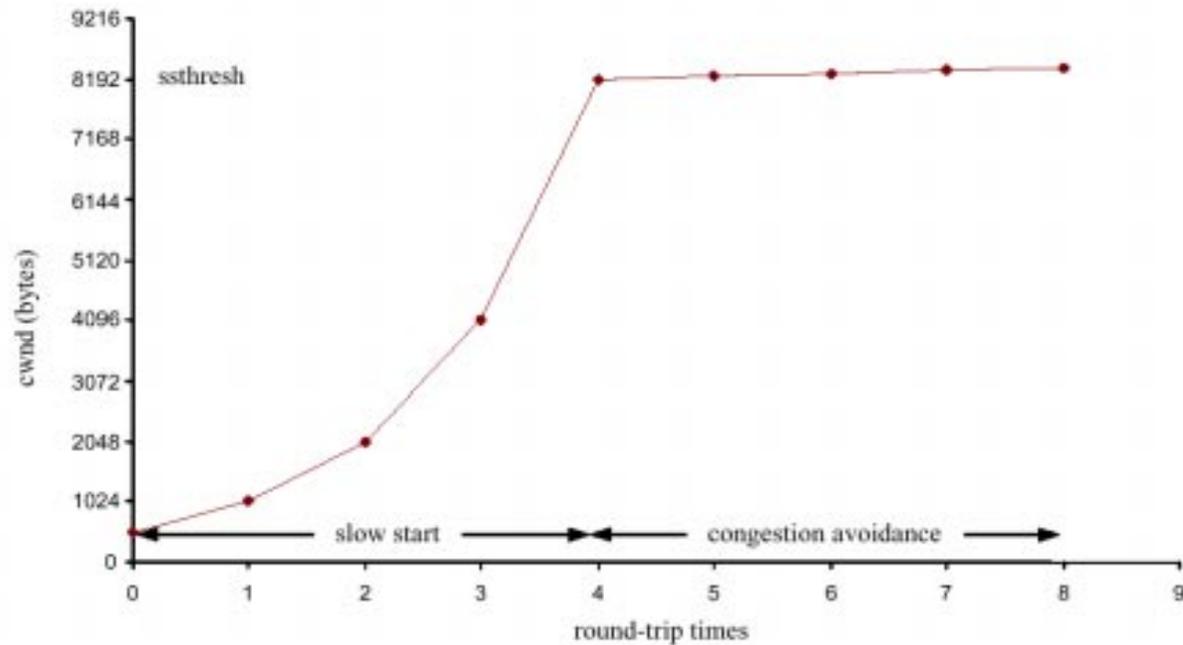
Figure 33: Graphical plot of congestion window (cwnd) as the connection goes from slow start to congestion avoidance behavior (figure from Mattias Ronquist, "TCP Reaction to Rapid Changes of the Link Characteristics due to Handover in a Mobile Environment", MS Thesis, Royal Institute of Technology, Teleinformatics, August 4, 1999.)

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 246 of 297
Internetworking/Internetteknik

Round-Trip Time Measurement

Fundamental to TCP's timeout and retransmission is the measurement (M) of the round-trip time (RTT). As the RTT changes TCP should modify its timeouts.

Originally TCP specified:

$$R \leftarrow \alpha R + (1 - \alpha)M$$

αa smoothing factor, with a recommended value of 0.9

$$RTO = R\beta$$

βa delay variance factor, with a recommended value of 2

RTO == retransmission timout time

Van Jacobson found that this could not keep up with wide fluctuations in RTT, which leads to more retransmissions, when the network is already loaded! So he proposed tracking the variance in RTT and gave formulas which compute the RTO based on the **mean** and **variance** in RTT and can be easily calculated using integer arithmetic (see Stevens, Vol. 1, pg. 300 for details).

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 247 of 297
Internetworking/Internetteknik

Karn's algorithm

When a retransmisson occurs, RTO is backed off, the packet retransmitted with the new longer RTO, and an ACK is received.

But is it the original ACK or the new ACK?

- we don't know, thus we don't recalculate a new RTO until an ACK is received for a segment which is not retransmitted

Congestion Avoidance Algorithm

Slow start keeps increasing cwnd, but at some point we hit a limit due to intervening routers and packets start to be dropped.

The algorithm assumes that packet loss means congestion[1]. Signs of packet loss:

- timeout occuring
- receipt of duplicate ACKs

Introduce another variable for each connection: ssthresh == slow start threshold

when data is acknowledged we increase cwnd:

- if cwnd < ssthreshold we are doing slow start; increases continue until we are half way to where we hit congestion before
- else we are doing congestion avoidance; then increase by 1/cwnd + 1/8 of segment size each time an ACK is received

(See Stevens, Vol. 1, figure 21.8, pg. 311 for a plot of this behavior)

---

1. Note: if your losses come from other causes (such as bit errors on the link) it will still think it is due to congestion!

Maguire
maguire@it.kth.se
TCP header continued
2006.03.13
TCP, HTTP, RPC, NFS, X 249 of 297
Internetworking/Internetteknik

Van Jacobson's Fast retransmit and Fast Recovery Algorithm

TCP is required to generate an immediate ACK (a duplicate ACK) when an out-of-order segment is received. This duplicate ACK should not be delayed. The purpose is to tell the sender that the segment arrived out of order and what segment number the receiver expects.

Cause:

- segments arriving out of order OR
- lost segment

If more than a small number (3) of duplicate ACKs are detected, assume that a segment has been lost; then retransmit the missing segment immediately (without waiting for a retransmission timeout) and perform congestion avoidance - but not slow start.

Why not slow start? Because the only way you could have gotten duplicate ACKs is if subsequent segments did arrive - which means that data is getting through.

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 250 of 297
Internetworking/Internetteknik

Per-Route Metrics

Newer TCPs keeps smoothed RTT, smoothed mean deviation, and slow start threshold in the routing table.

When a connection is closed, if there was enough data exchanged (defined as 16 windows full) - then record the parameters

- i.e., 16 RTT samples $\Rightarrow$ smoothed RTT is accurate to ~5%

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 251 of 297
Internetworking/Internetteknik

TCP Persist Timer

If the window size is 0 and an ACK is lost, then receiver is waiting for data and sender is waiting for a non-zero window!

To prevent deadlock, introduce a **persist timer** that causes sender to query the receiver periodically with **window probes** to find out if window size has increased.

Window probes sent every 60 seconds - TCP never gives up sending them.

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 252 of 297
Internetworking/Internetteknik

TCP Keepalive Timer

No data flows across an idle TCP connection - connections can persist for days, months, etc. Even if intermediate routers and links go down the connection persists!

However, some implementations have a **keepalive timer.**

Host Requirements RFC gives 3 reasons **not** to use keepalive messages:

- can cause perfectly good connections to be dropped during transient failures
- they consume unnecessary bandwidth
- they produce additional packet charges (if you are on a net that charges per packet)

Host Requirements RFC says you can have a keepalive time but:

- it must not be enabled unless an application specifically asks
- the interval must be configurable, with a default of no less than 2 hrs.

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 253 of 297
Internetworking/Internetteknik

TCP Performance

TCP's path MTU discovery:

- use **min**(MTU of outgoing interface, MSS announced by other end)
- use per-route saved MTU
- once an initial segment size is chosen - all packets have don't fragment bit set
- if you get an ICMP "Can't fragment" message - recompute Path MTU.
- periodically check for possibility of using a larger MTU (RFC 1191 recommends 10 minute intervals)

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 254 of 297
Internetworking/Internetteknik

Long Fat Pipes

Networks with large bandwidth-delay products are called **Long Fat Networks (LFNs)** - pronounced "elefants".

TCP running over a LFN is a **Long Fat Pipe.**

- Window Scale option - to avoid 16 bit window size limit
- Timestamp option - putting a time stamp in each segment allows better computation of RTT
- Protection Against Wrapped Sequence Numbers (PAWS) - with large windows you could have sequence number wrap around and not know which instance of a given sequence number is the correct one; solved by using timestamps (which must simply be monotonic)
- T/TCP - TCP extension for Transactions; to avoid the three way handshake on connection setup and shorten the TIME_WAIT state. (for details of T/TCP see Stevens, Vol.3)

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 255 of 297
Internetworking/Internetteknik

Measuring TCP Performance

Measured performance:

- Performance on Ethernets at ~90% of theoretical value (using workstations)
- TCP over FDDI at 80-98% of theoretical value
- TCP (between two Crays) at 781 Mbits/sec over a 800Mbit/sec HIPPI channel
- TCP at 907Mbits/sec on a loopback interface of a Cray Y-MP.

Practical limits:

- can't run faster than the slowest link
- can't go faster than the memory bandwidth of the slowest machine (since you have to touch the data at least once)
- you can't go faster than the window size offered by the receiver divided by the round trip time (comes from the calculation of the bandwidth delay product)
  - thus with the maximum window scale factor (14) $\Rightarrow$ window size of 1.073 Gbits; just divide by RTT to find the maximum bandwidth

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 256 of 297
Internetworking/Internetteknik

Example of TCP behavior [1]
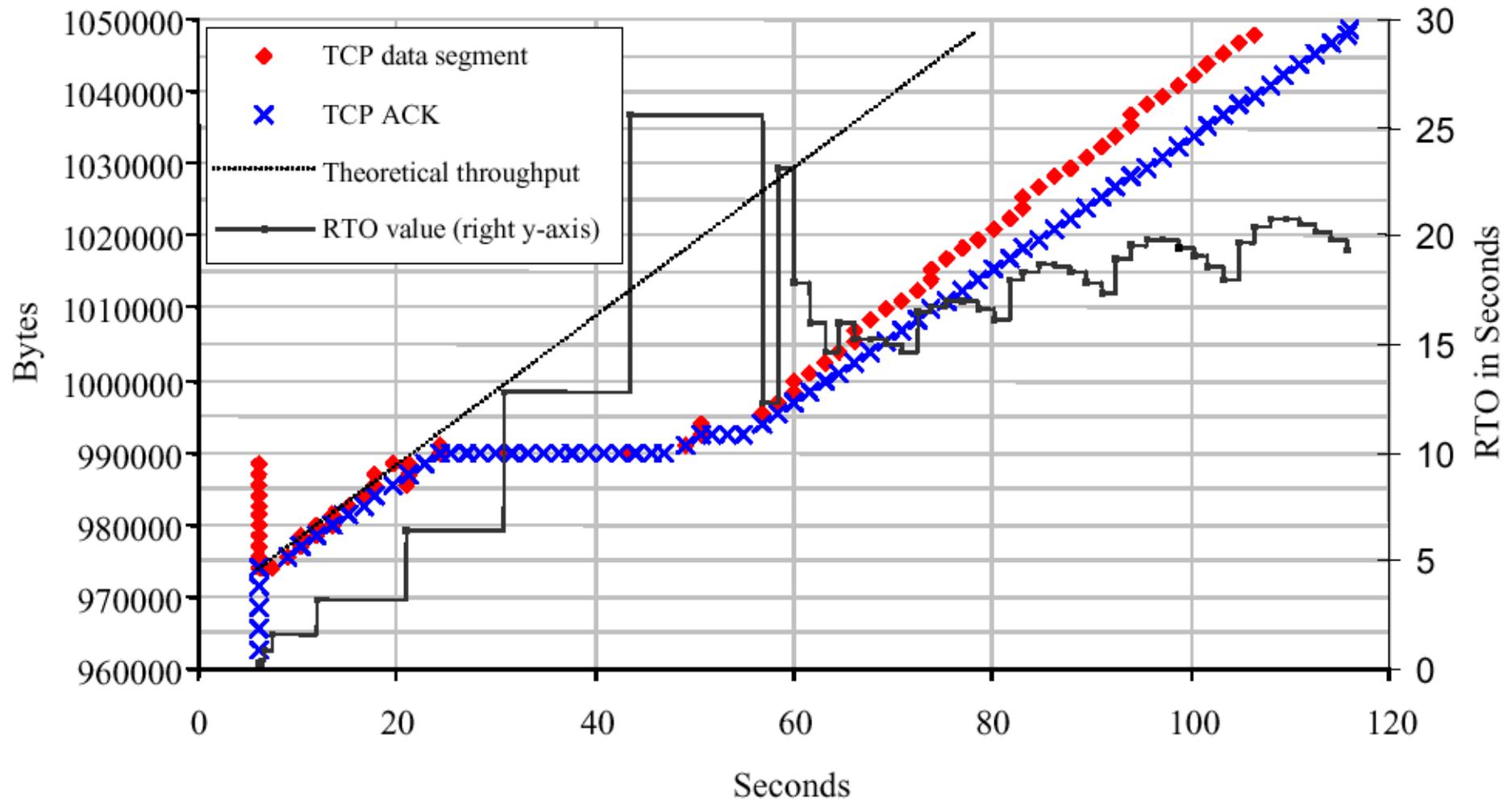
Figure 7-3: TCP segments and ACKs collected at the correspondent host after the handover. The RTO values and the theoretical throughput of the PPP link are also included.

---

1. Figure 7-3, from Mattias Ronquist, "TCP Reaction to Rapid Changes of the Link Characteristics due to Handover in a Mobile Environment", MS Thesis, Royal Institute of Technology, Teleinformatics, August 4, 1999, p.38.

Maguire
maguire@it.kth.se

TCP header continued
2006.03.13

TCP, HTTP, RPC, NFS, X 257 of 297
Internetworking/Internetteknik

TCP servers

Stevens, Vol. 1, pp. 254-260 discusses how to design a TCP server, which is similar to list of features discussed for UDP server, but now it is incoming connection requests which are queued rather than UDP datagrams

- note that incoming requests for connections which exceed the queue - are silently ignored - it is up to the sender to time out it active open
- this limited queuing has been one of the targets of denial of service attacks
  - TCP SYN Attack - see *http://cio.cisco.com/warp/public/707/4.html*
  - Increase size of the SYN_RCVD queue (kernel variable somaxconn limits the maximum backlog on a listen socket - backlog is the sum of both the SYN_RCVD and accept queues) and decrease the time you will wait for an ACK in response to your SYN_ACK
  - for a nice HTTP server example, see
    http://www.cs.rice.edu/CS/Systems/Web-measurement/paper/node3.html

Maguire
maguire@it.kth.se
TCP servers
2006.03.13
TCP, HTTP, RPC, NFS, X 258 of 297
Internetworking/Internetteknik

Hypertext Transfer Protocol (HTTP)

This protocol is the basis for the World Wide Web (WWW).

Uses TCP connections. HTTP traffic growing at a very high rate - partly due to popularity and partly due to the fact that it can easily include text, pictures, movies, … .
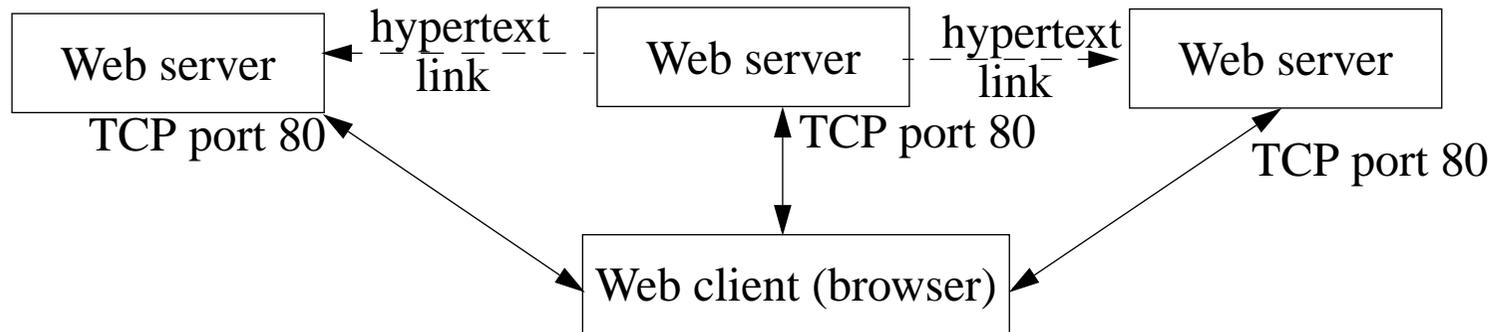
Figure 34: Organization of a Web client-server
(see Stevens, Vol. 3, figure 15.1, pg. 210)

HTTP described by an Internet Draft in 1993; replaced with RFC 1945, "*Hypertext Transfer Protocol -- HTTP/1.0*", May 1996; RFC 2068, "*Hypertext Transfer Protocol -- HTTP/1.1*", January 1997; replaced by RFC 2616, June 1999, RFC 2817 "*Upgrading to TLS Within HTTP/1.1*", May 2000.

Maguire
maguire@it.kth.se

Hypertext Transfer Protocol (HTTP)
2006.03.13

TCP, HTTP, RPC, NFS, X 259 of 297
Internetworking/Internetteknik

HTTP Prochocol

2 message types:

- request

HTTP 1.0 request

request line

headers (0 or more)

<blank_line>

body (only for a POST request)

- response

HTTP/1.0 response

status line

headers (0 or more)

<blank_line>

body

Maguire
maguire@it.kth.se

Hypertext Transfer Protocol (HTTP)
2006.03.13

TCP, HTTP, RPC, NFS, X 260 of 297
Internetworking/Internetteknik

HTTP Requests

request-line == request request-URI HTTP-version

Three requests:

- GET - returns information identified by request URI
- HEAD - similar to GET but only returns header information
- POST - sends a body with a request; used for posting e-mail, news, sending a fillin form, etc.

Universal Resource Idendifiers (URIs) - described in RFC 1630, URLs in RFC 1738 and RFC 1808.

status-line == HTTP-version response-code response-phrase

Maguire
maguire@it.kth.se

Hypertext Transfer Protocol (HTTP)
2006.03.13

TCP, HTTP, RPC, NFS, X 261 of 297
Internetworking/Internetteknik

HTTP Header fields

HTTP Header names (See Stevens, figure 13.3, Vol. 3, pg. 166)

Header name	Request?	Response?	Body?
Allow			
Authorization			
Content-Encoding			
Content-Length			
Content-Type			
Date			
Expires			
From			
If-Modified-Since			
Last-Modified			
Location			
MIME-Version			
Pragma			
Referer			
Server			
User-Agent			
WWW-Authenticate			

Maguire
maguire@it.kth.se

Hypertext Transfer Protocol (HTTP)
2006.03.13

TCP, HTTP, RPC, NFS, X 262 of 297
Internetworking/Internetteknik

HTTP Response Codes

HTTP 3-digit response code (See Stevens, figure 13.4, Vol. 3, pg. 167)

Response	Description
1yz	Informational. Not currently used
	Success
200	OK, request succeeded.
201	OK, new resource created (POST command)
202	Request accepted but processing not completed
204	OK, but no content to return
	Redirection; further action needs to be taken by user agent
301	Requested resource has been assigned a new permanent URL
302	Requested resource resides temporarily under a different URL
304	Document has not been modified (conditional GET)
	Client error
400	Bad request
401	Unauthorized; request requires user authentication
403	Forbidden for unspecified reason
404	Not found
	Server error
500	Internal server error
501	Not implemented
502	Bad gateway; invalid response from gateway or upstream server
503	Service temporarily unavailable

Maguire
maguire@it.kth.se

Hypertext Transfer Protocol (HTTP)
2006.03.13

TCP, HTTP, RPC, NFS, X 263 of 297
Internetworking/Internetteknik

Client Caching

Client can cache HTTP documents along with the date and time the document was fetched.

If the document is cached, then the If-Modified-Since header can be sent to check if the document has changed since the copy was cached - thus saving a transfer - but costing a round trip time and some processing time. This is called a conditional GET.

Maguire
maguire@it.kth.se

Hypertext Transfer Protocol (HTTP)
2006.03.13

TCP, HTTP, RPC, NFS, X 264 of 297
Internetworking/Internetteknik

Server Redirect

Response code 302, along with a new location of the request-URI.

Maguire
maguire@it.kth.se

Hypertext Transfer Protocol (HTTP)
2006.03.13

TCP, HTTP, RPC, NFS, X 265 of 297
Internetworking/Internetteknik

Multiple simultaneous connections to server

GET of a page with multiple objects on it (such as GIF images) - one new connection for each object, all but the first can occur in parallel!
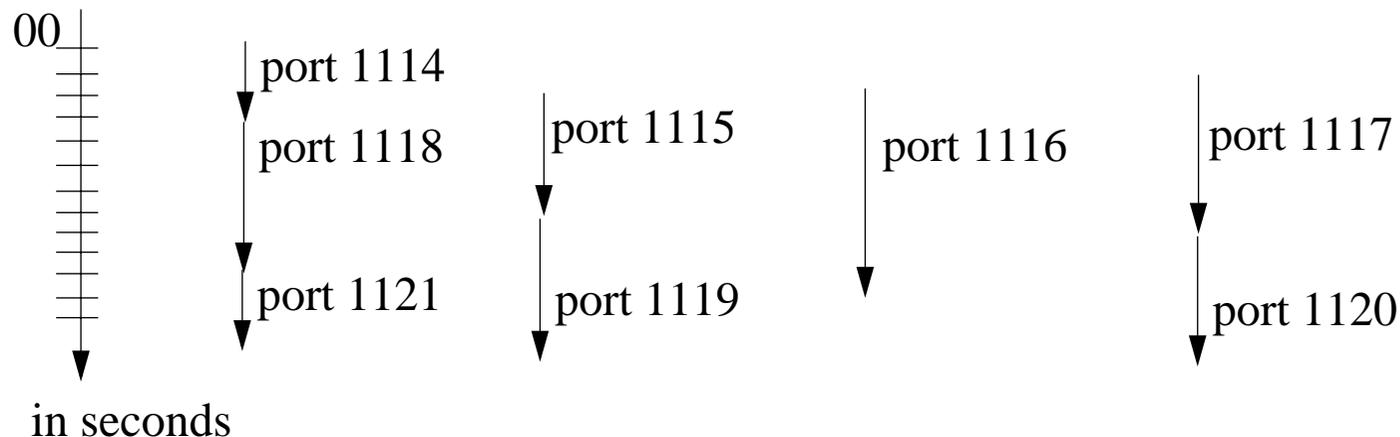
Figure 35: Timeline of eight TCP connection for a home page and seven GIF images
(see Stevens, Vol. 3, figure 113.5, pg. 171)

Note that the port 1115, 1116, and 1117 requests start before 1114 terminates, Netscape can initiate 3 non-blocking connects after reading the end-of-file but before closing the first connection.

Maguire
maguire@it.kth.se

Hypertext Transfer Protocol (HTTP)
2006.03.13

TCP, HTTP, RPC, NFS, X 266 of 297
Internetworking/Internetteknik

Decrease in total time to produce a response:

(from Stevens, figure 13.6, Vol. 3, pg. 171)

Simultaneous connections	Total time (seconds)
1	14.5
2	11.4
3	10.5
4	10.2
5	10.2
6	10.2
7	10.2

Why no improvement beyond 4?

- program has an implementation limit of 4, even if you specify more!
- gains beyond 4 are probably small (given the small difference between 3 and 4) - but Steven's has not checked!

Maguire
maguire@it.kth.se

Hypertext Transfer Protocol (HTTP)
2006.03.13

TCP, HTTP, RPC, NFS, X 267 of 297
Internetworking/Internetteknik

Problems with multiple connections

Such multiple connection have problems:

- Unfair to other protocols (such as FTP) which are using one connection at a time to fetch multiple files

- Congestion information is not passed to the other connections

- can more easily overflow the server's incomplete connection queue which can lead to large delays as the host retransmits SYNs.

  - ◆ In fact it looks like a denial of service attack -- which is trying to flood the host!

Maguire
maguire@it.kth.se

Problems with multiple connections
2006.03.13

TCP, HTTP, RPC, NFS, X 268 of 297
Internetworking/Internetteknik

HTTP Statistics

Statistics for individual HTTP connections (see Stevens, figure 13.7, Vol. 3, pg. 172)

	Median	Mean
client bytes/connection	224	266
server bytes/connection	3,093	7,900
connection duration (seconds)	3.4	22.3

Mean is often skewed by very large transfers.

Maguire
maguire@it.kth.se

Problems with multiple connections
2006.03.13

TCP, HTTP, RPC, NFS, X 269 of 297
Internetworking/Internetteknik

What happens when you make an HTTP request

connect to www.it.kth.se

makeup UDP packet for DNS lookup

choose name server

choose LAN adapter

send ARP for name server or gateway MAC

request

reply

Reality!

Logically

send DNS query

request *

reply

send ARP for HTTP server's MAC or send via gateway MAC

send TCP SYN to IT's IP address, via local router's MAC address
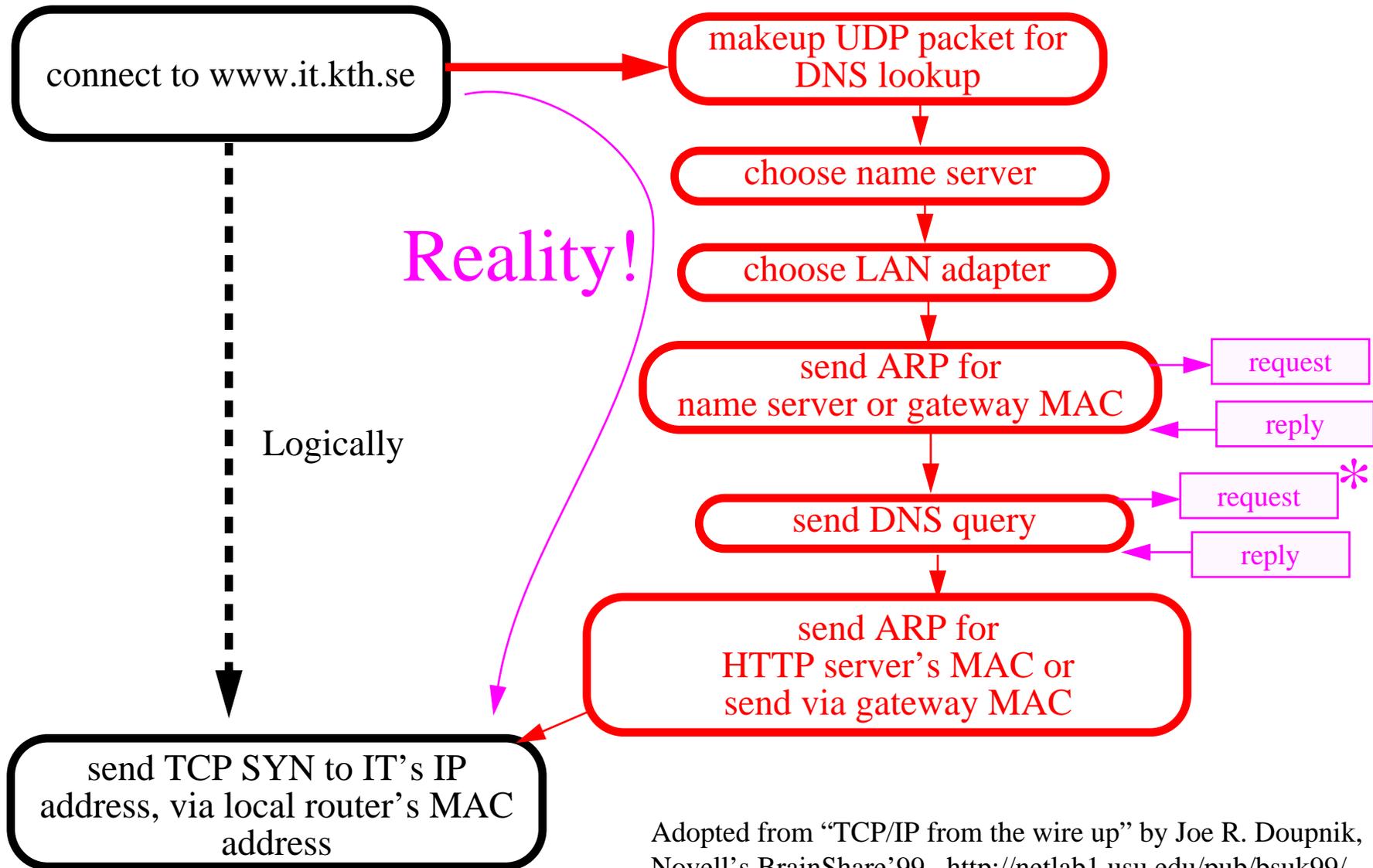
Adopted from "TCP/IP from the wire up" by Joe R. Doupnik, Novell's BrainShare'99. http://netlab1.usu.edu/pub/bsuk99/

Maguire
maguire@it.kth.se

Problems with multiple connections
2006.03.13

TCP, HTTP, RPC, NFS, X 270 of 297
Internetworking/Internetteknik

HTTP Performance Problems

HTTP opens one connection for **each** document.

- Each such connection involves slow start - which adds to the delay
- Each connection is normally closed by the HTTP server - which has to wait TIME_WAIT, thus lots of control blocks are waiting in the server.

Proposed changes:

- have client and server keep a TCP connection open {this requires that the size of the response (Content-Length) be generated}
  - requires a change in client and server
  - new header Pragma: hold-connection
- GETALL - causes server to return document and all in-lined images in a single response
- GETLIST - similar to a client issuing a series of GETs
- HTTP-NG (aka HTTP/1.1) - a single TCP connection with multiple sessions {it is perhaps the first TCP/IP session protocol}
  - HTTP/1.1 also has another feature - the server knows what hostname was in the request, thus a single server at a single IP address can be the HTTP server under many "names" - hence providing "Web hotel" services for many firms _but_ only using a single IP address.

Maguire
maguire@it.kth.se

Problems with multiple connections
2006.03.13

TCP, HTTP, RPC, NFS, X 271 of 297
Internetworking/Internetteknik

HTTP performance

Joe Touch, John Heidemann, and Katia Obraczka, "Analysis of HTTP Performance",USC/Information Sciences Institute, August 16, 1996, Initial Release, V1.2 -- `http://www.isi.edu/lsam/publications/http-perf/`

John Heidemann, Katia Obraczka, and Joe Touch, "Modeling the Performance of HTTP Over Several Transport Protocols", IEEE/ACM Transactions on Networking 5(5), October 1997. November, 1996.

`http://www.isi.edu/~johnh/PAPERS/Heidemann96a.html`

Simon E Spero, "Analysis of HTTP Performance problems"

`http://sunsite.unc.edu/mdma-release/http-prob.html` This is a nice introduction to HTTP performance.

John Heidemann, "Performance Interactions Between P-HTTP and TCP Implementations". ACM Computer Communication Review, 27 2, 65-73, April, 1997. `http://www.isi.edu/~johnh/PAPERS/Heidemann97a.html`

Maguire
maguire@it.kth.se

HTTP performance
2006.03.13

TCP, HTTP, RPC, NFS, X 272 of 297
Internetworking/Internetteknik

Web Enabled Devices

emWare - thin client (30 bytes of RAM, 750 bytes of ROM) - for a very thin client

URL: *http://www.emware.com/*

Splits the web server into a very tiny server on the device and more processing (via applets) in the desktop system (where the WEB browser is running).

Axis Communications AB - *http://www.axis.com* - produces many web enabled devices - from thin clients to "cameras" running an embedded Linux

Maguire
maguire@it.kth.se

Web Enabled Devices
2006.03.13

TCP, HTTP, RPC, NFS, X 273 of 297
Internetworking/Internetteknik

Network File System (NFS)

NFS is based on Sun's Remote Procedure Call (RPC) - RFC 1831

- from the caller's point of view it looks much like a local procedure call
- from the callee's point of view it seems much like a local procedure call
- Request-reply protocol
- UDP or TCP transport
- Standardized data representation - RPC encodes its data using the eXternal Data Representation (XDR) protocol, RFC 1832
- Authentication {for example, for NFS operations, authentication usually based on relaying UNIX user and group IDs to the file server for permission checking}

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 274 of 297
Internetworking/Internetteknik

Remote Procedure Call (RPC)

Two versions:

- using Sockets API and works with TCP and UDP
- using TLI API TI-RPC (Transport Independent) and works with any transport layer provided by the kernel

Format of an RPC call message as a UDP datagram

IP Header	20 bytes
UDP Header	8
transaction ID (XID)	4
call (0)	4
RPC version (2)	4
program number	4
version number	4
procedure number	4
credentials	upto 400 bytes
verifier	upto 400 bytes
procedure parameters	N

- XID set by client and returned by server (client uses it to match requests and replies)

Maguire
maguire@it.kth.se
Network File System (NFS)
2006.03.13
TCP, HTTP, RPC, NFS, X 275 of 297
Internetworking/Internetteknik

- program number, program version, procedure number identifies the procedue to be called
- credentials identify the client - sometimes the user ID and group ID
- verifier - used with secure RPC (to identify the server); uses DES encryption

1

Format of an RPC reply message as a UDP datagram

IP Header	20 bytes
UDP Header	8
transaction ID (XID)	4
call (1)	4
status (0=accepted)	4
verifier	upto 400 bytes
procedure results	N

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 276 of 297
Internetworking/Internetteknik

External Data Representation (XDR)

used to encode value in RPC messages - see RFC 1014

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 277 of 297
Internetworking/Internetteknik

Port Mapper

RPC server programs use ephemeral ports - thus we need a well known port to be able to find them

Servers register themselves with a registrar - the **port mapper**
(called rpcbind in SVR4 and other systems using TI-RPC)

Port mapper is at well know port: 111/UDP and 111/TCP

The port mapper is an RPC server with program number 100000, version 2, a TCP port of 111, a UDP port of 111.

Servers register themselves with RPC calls and clients query with RPC calls:

- PMAPPROC_SET - register an entry
- PMAPPROC_UNSET - unregister an entry
- PMAPPROC_GETPORT - get the port number of a given instance
- PMAPPROC_DUMP - returns all entries (used by "rpcinfo -p")

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 278 of 297
Internetworking/Internetteknik

NFSspy

Insert a new pointer in place of the RPC we want to snoop

Embed this earlier code in our code:

Before   After
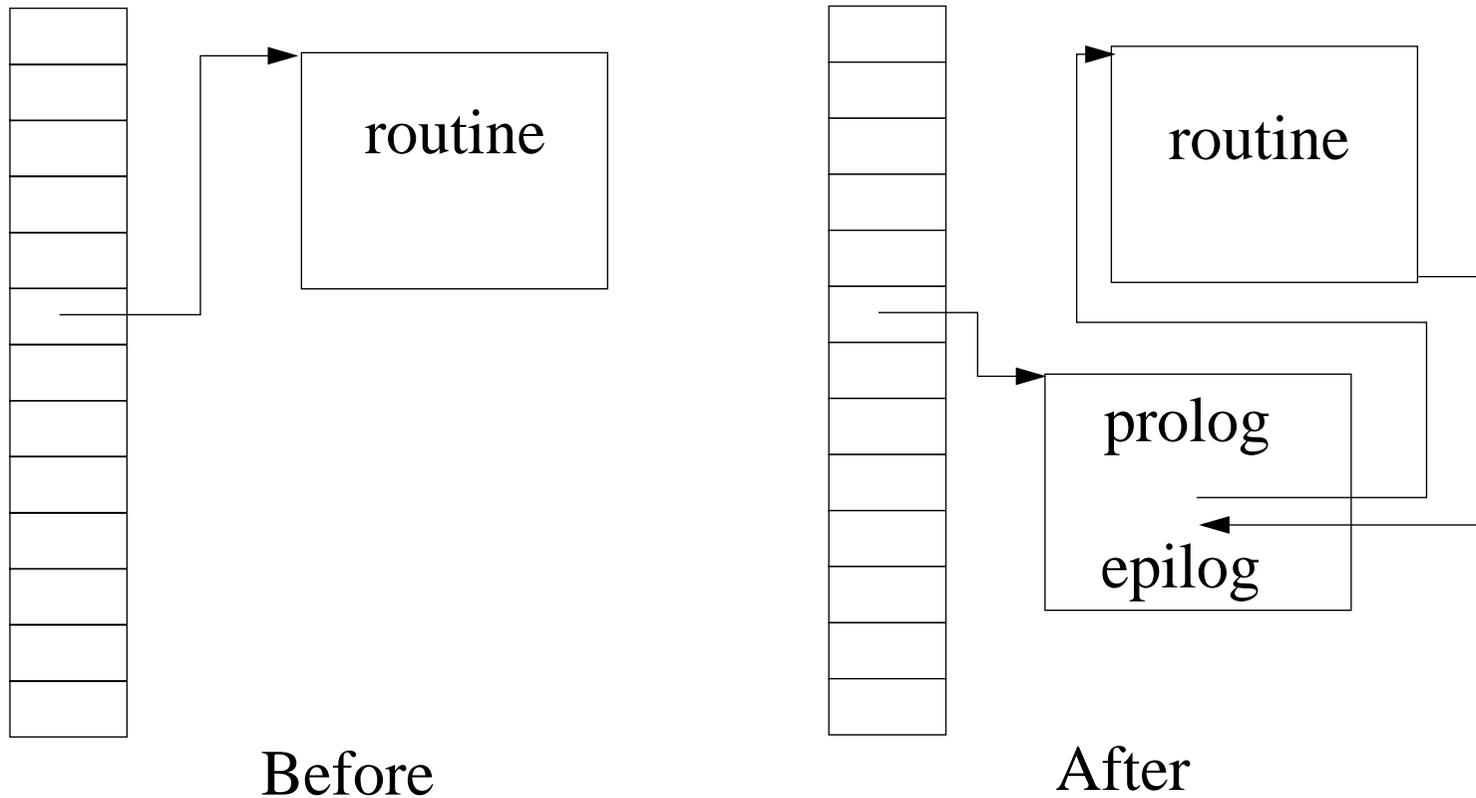
NFSspy problem

Imagine several students each insert a new pointer in place of the RPC they want to snoop:

What happens if this one dies?

routine

prolog   prolog   prolog

epilog   epilog   epilog
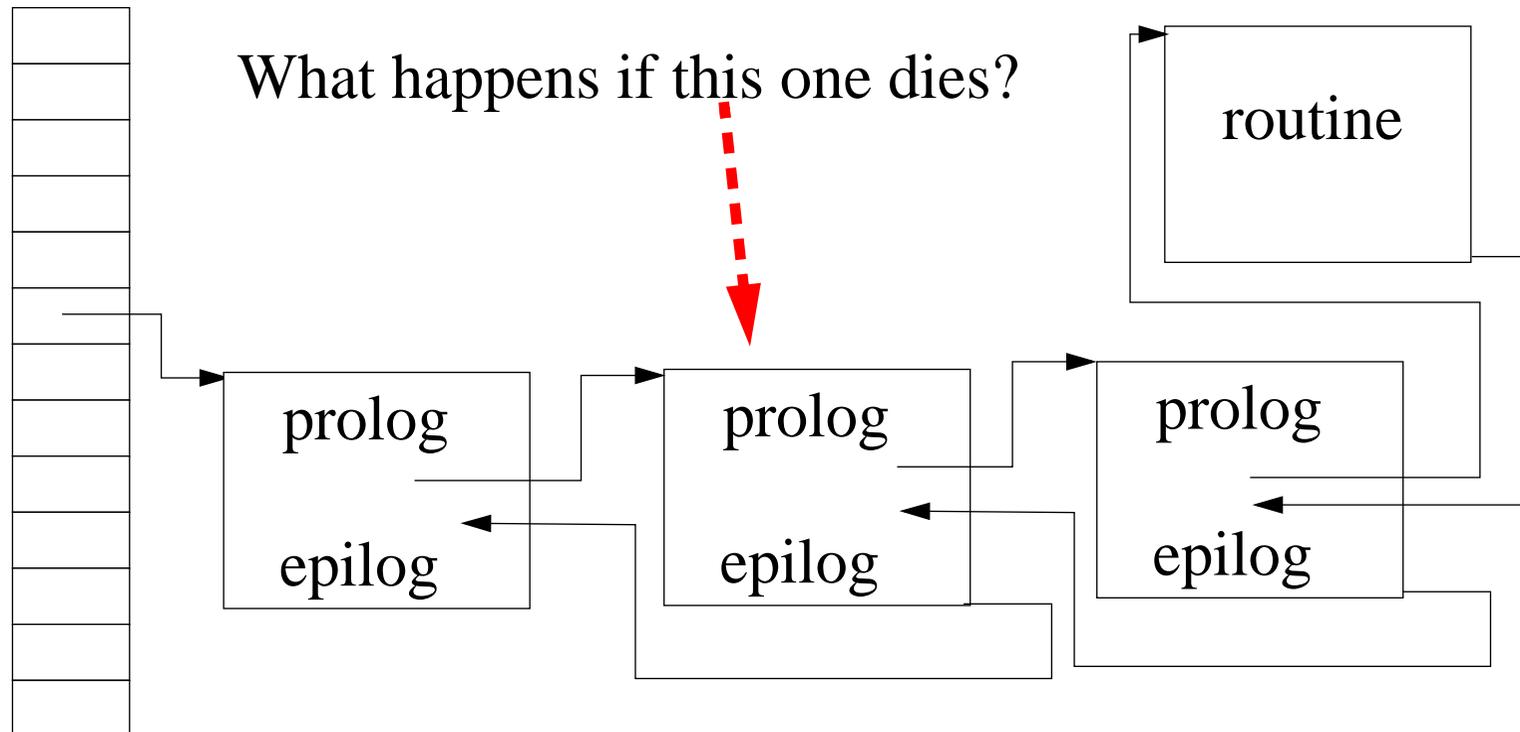
The routine is no longer callable by anyone!

nfsspy

Initial implementations were written by Seth Robertson, Jon Helfman, Larry Ruedisueli, Don Shugard, and other students for a project assignment in my course on Computer Networks at Columbia Univ. in 1989. There is a report about one implementation by Jon Helfman, Larry Ruedisueli, and Don Shugard, "Nfspy: A System for Exploring the Network File System", AT&T Bell Laboratories, 11229-890517-07TM.

See also "NFS Tracing By Passive Network Monitoring" by Matt Blaze, ~1992,
`http://www.funet.fi/pub/unix/security/docs/papers/nfsspy.ps.gz`

Matt's program builds upon an rpcspy program and this feeds packets to his nfstrace program and other scripts.

Seth Robertson's version even inverted the file handles to show the actual file names.

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 281 of 297
Internetworking/Internetteknik

NFS protocol, version 2 (RFC 1094)

- provides transparent file access
- client server application built on RPC

Generally, NFS server at 2049/UDP - but it can be at different ports
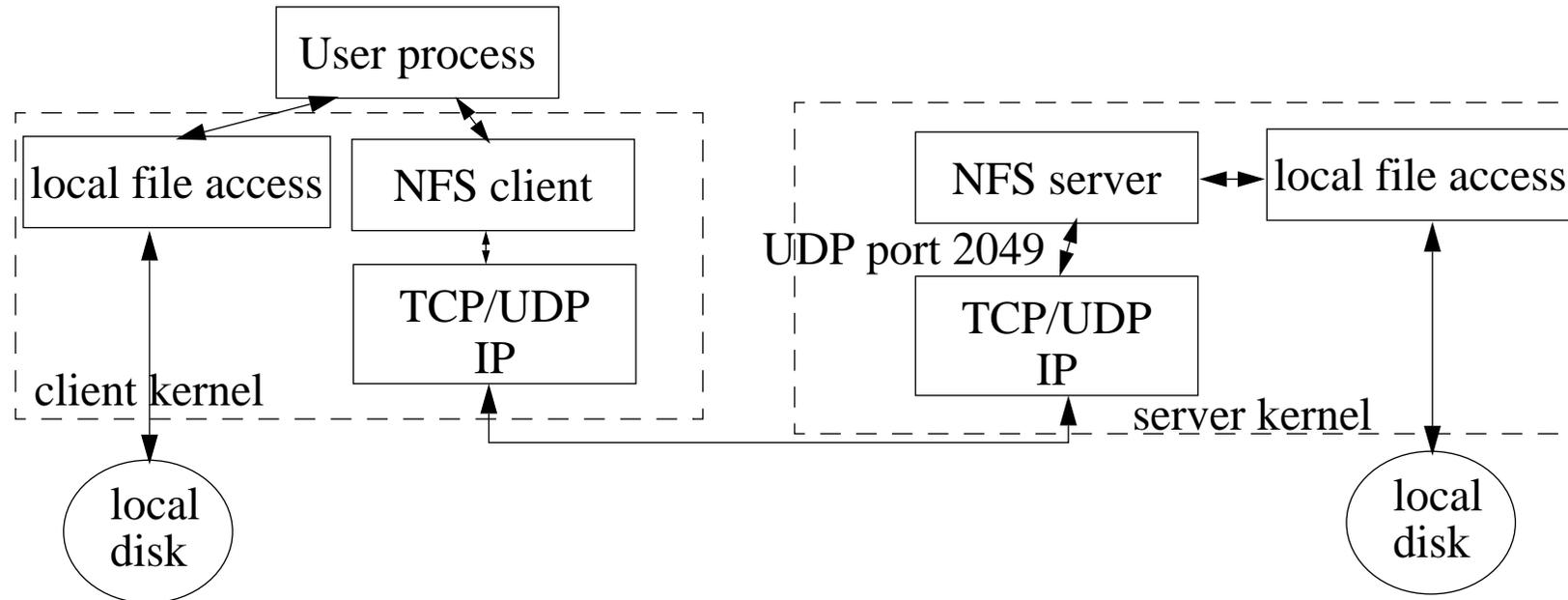
Figure 36: NFS client and NFS server (see Stevens, Vol. 1, figure 29.3, pg. 468)

Most NFS servers are multithreaded - so that multiple requests can be in process at one time. If the server kernel does not support mutlithreading then multiple server processes ("nfsd") are used.

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 282 of 297
Internetworking/Internetteknik

Often there are multiple NFS clients ("biod") running on the client machine - each processes one call and waits inside the kernel for the reply.

NFS consists of more than just the NFS protocol

Various RPC programs used with NFS (see Stevens, Vol. 1, pg. 469)

Application	program number	version numbers	Number of procedures
port mapper	100000	2	4
NFS	100003	2	15
mount	100005	1	5
lock manager	100021	1,2,3	19
status monitor	100024	1	6

lock manager and status monitor allow locking of portions of files

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 283 of 297
Internetworking/Internetteknik

NFS File Handles

To reference a file via NFS we need a **file handle,** an opaque object used to reference a file or directory on the server.

File handle is created by the server - upon an lookup; subsequent client requests just simply pass this file handle to the appropriate procedure (they never look at the contents of this object - hence it is opaque).

- in version 2, a file handle is 32 bytes
- in version 2, a file handle is 64 bytes

UNIX systems generally encode the filesystem ID (major and minor dev numbers), the i-node number, and an i-node generation number into the file handle.

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 284 of 297
Internetworking/Internetteknik

NFS Mount protocol

Server can check IP address of client, when it gets a mount command from a client to see if this client is allowed to mount the given filesystem; Mount daemon returns the file handle of the given filesystem.

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 285 of 297
Internetworking/Internetteknik

NFS Procedures

Procedure	Description
NFSPROC_GETATTR	return the attributes of a file
NFSPROC_SETATTR	set the attributes of a file
NFSPROC_STATFS	return the status of a filesystem
NFSPROC_LOOKUP	lookup a file - returns a file handle
NFSPROC_READ	read from a file, starting at specified offset for n bytes (upto 8192 bytes)
NFSPROC_WRITE	Write to a file, starting at specified offset for n bytes (upto 8192 bytes) Writes are synchronous - i.e., server responds OK when file is actually written to disk (this can often be changed as an option at mount time - but you can get into trouble)
NFSPROC_CREATE	Create a file
NFSPROC_REMOVE	Delete a file
NFSPROC_RENAME	Rename a file
NFSPROC_LINK	make a hard link to a file
NFSPROC_SYMLINK	make a symbolic link to a file
NFSPROC_READLINK	return the name of the file to which the symbolic link points
NFSPROC_MKDIR	create a directory
NFSPROC_RMDIR	delete a directory
NFSPROC_READDIR	read a directory

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 286 of 297
Internetworking/Internetteknik

NFS over TCP

Provided by some vendors for use over WANs.

- All applications on a given client share the same TCP connection.
- Both client and server set TCP keepalive timers
- If client detects that server has crashed or been rebooted, it tries to reconnect to the server
- if the client crashes,the client gets a new connection, the keepalive timer will terminate the half-open former connection

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 287 of 297
Internetworking/Internetteknik

NFS Statelessness

NFS is designed to be stateless

- the server does not keep track of what clients are accessing which files
- there are no open or close procedures; just LOOKUP
- being stateless simplifies server crash recovery
- clients don't know if the server crashes
- only the client maintains state

Most procedures (GETATTR, STATFS, LOOKUP, READ, WRITE, READIR) are idempotent (i.e., can be executed more than once by the server with the same result).

Some (CREATE, REMOVE, RENAME, SYMLINK, MKDIR, RMDIR) are not. SETATTR is idempotent unless it is truncating a file.

To handle non-idempotent requests - most servers use recent-reply cache, checking their cache to see if they have already performed the operation and simply return the same value (as before).

Maguire
maguire@it.kth.se

Network File System (NFS)
2006.03.13

TCP, HTTP, RPC, NFS, X 288 of 297
Internetworking/Internetteknik

X Window System

- Client-server application that lets multiple clients share a bit-mapped display.

- One server manages the display, keyboard, mouse, …

- X requires a reliable bidirectional bitstream protocol (such as TCP).

- The server does a passive open on port 6000+n, where n is the display number (usually 0)

- X can also use UNIX domain sockets
  (with the name /tmp/.X11-unix/Xn, where n is the display number)

- > 150 different messages in the X protocol (for details see Nye, 1992)
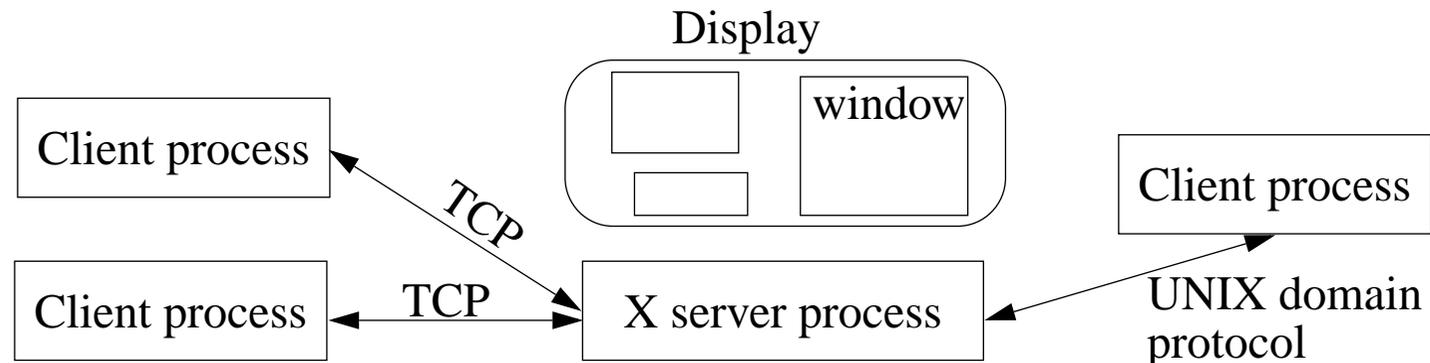
Figure 37: Clients using a X server to access one display

Maguire
maguire@it.kth.se
X Window System
2006.03.13
TCP, HTTP, RPC, NFS, X 289 of 297
Internetworking/Internetteknik

- All clients (even those on different hosts) communicate with the same server.

- Lots of data can be exchanged between client and server
  - xclock - send date and time once per second
  - Xterm - send each key stroke (a 32 byte X message $\Rightarrow$ 72 bytes with IP and TCP headers)
  - some applications read and write entire 32 bit per pixel images in cine mode from/to a window!

Maguire
maguire@it.kth.se

X Window System
2006.03.13

TCP, HTTP, RPC, NFS, X 290 of 297
Internetworking/Internetteknik

Low Bandwidth X

X was optimized for use across LANs.

For use across low speed links - various techniques are used:

- caching
- sending differences from previous packets
- compression, …

Maguire
maguire@it.kth.se

X Window System
2006.03.13

TCP, HTTP, RPC, NFS, X 291 of 297
Internetworking/Internetteknik

Xscope

Interpose a process between the X server and X client to watch traffic.

For example, xscope could be run as if it were display "1", while passing traffic to and from display "0". See Stevens, Vol.1, pp. 488-489 for more details (or try running it!)

J.L. Peterson. XSCOPE: A Debugging and Performance Tool for X11. Proceedings of the IFIP 11th World Computer Congress, September, 1989, pp. 49-54.

See also XMON - An interactive X protocol monitor

Both are available from: *ftp://ftp.x.org/pub/R5/*

Maguire
maguire@it.kth.se

X Window System
2006.03.13

TCP, HTTP, RPC, NFS, X 292 of 297
Internetworking/Internetteknik

Additional tools for watching TCP

Program	Description
IPerf	Measure bandwidth availabity using a client and server. Determines total bandwidth, delay jitter, loss, determine MTU, support TCP window size, …
Pathchar	Determine per hop network path characteristics (bandwidth, propagation delay, queue time and drop rate. It utilizes a series of packets with random payload sizes over a defined period of time to each hop in a path.
Pchar	Updated version of Pathchar -- by Bruce Mah
Netlogger	NetLogger includes tools for generating precision event logs that can be used fpr detailed end-to-end application & system level monitoring, and tools for visualizing log data to view the state of a distributed system in real time.
Treno	Measure single stream bulk transfer capacity. TReno doesn't actually use TCP slow start, but instead emulates it. It actually sends UDP packets to unused ports and uses the returned error meassages to determine the packet timing.
Mping	Measure queuing properties during heavy congestion
tdg	produce graphs of TCP connections from tcpdump files, suitable for use with xgraph. A perl script which produces time-sequence plots from tcpdump files.

Maguire
maguire@it.kth.se
Additional tools for watching TCP
2006.03.13
TCP, HTTP, RPC, NFS, X 293 of 297
Internetworking/Internetteknik

Program	Description
tcptrace	parses raw tcpdump files to extract information and xplot files.
xplot	generate graphs from plot data in X Windows.
testrig	automated connection diagnosis tool; generates a test flow & display a time sequence plot based on that flow.
aspath	determine traffic usage by AS path.

Maguire
maguire@it.kth.se

Additional tools for watching TCP
2006.03.13

TCP, HTTP, RPC, NFS, X 294 of 297
Internetworking/Internetteknik

Transaction TCP (T/TCP)

Piggyback a query in the TCP open - so that you don't have to wait a long time for sending a query / response

Maguire
maguire@it.kth.se

Transaction TCP (T/TCP)
2006.03.13

TCP, HTTP, RPC, NFS, X 295 of 297
Internetworking/Internetteknik

Summary

This lecture we have discussed:

- TCP
- HTTP
- Web enabled devices
- RPC, XDR, and NFS
- X Window System, and
- some tools for looking at these protocols

Maguire
maguire@it.kth.se

Summary
2006.03.13

TCP, HTTP, RPC, NFS, X 296 of 297
Internetworking/Internetteknik

References

[23] Information Sciences Institute, University of Southern California, Transmission Control Protocol, IETF, RFC 793, September 1981

Maguire
maguire@it.kth.se

References
2006.03.13

TCP, HTTP, RPC, NFS, X 297 of 297
Internetworking/Internetteknik

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 6: SCTP

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapter 13

KTH Information and Communication Technology

Transport layer protocols

- ## User Datagram Protcol (UDP)
  - Connectionless unreliable service

- ## Transmission Control Protocol (TCP)
  - Connection-oriented reliable stream service

- ## Stream Control Transmission Protocol (SCTP) <<< today's topic
  - Reliable message oriented service - a modern transmission protocol

Maguire
maguire@it.kth.se

Transport layer protocols
2006.03.13

SCTP 299 of 342
Internetworking/Internetteknik

Stream Control Transmission Protocol (SCTP) [26]

Provides a reliable message-oriented service; combining best of TCP & UDP.

- SCTP utilizes full-duplex associations

- SCTP applications write messages to one of several streams and read messages from these streams
    - each unit is a **chunk**
    - here are record makers $\Rightarrow$ the receiver **can** tell how much the sender wrote into the stream at any given time
    - multiple streams prevents a loss on one stream from affecting other streams

- SCTP supports multihoming
    - the sender and receiver can utilize multiple interfaces with multiple IP addresses $\Rightarrow$ increased fault tolerance
    - current implementations do **not** support *load balancing* (i.e., only supports failover)

- SCTP provides reliability
    - via acknowledgements, timeouts, retransmission, …

- SCTP provides flow control

- SCTP tries to avoid causing congestion

Maguire
maguire@it.kth.se

Stream Control Transmission Protocol (SCTP) [26]
2006.03.13

SCTP 300 of 342
Internetworking/Internetteknik

SCTP Applications

- Initial goal of IETF Sigtran WG was to support SS7 applications over IP
  - For example, SMS transfer!
  - For an example see [24], [28]

- new applications being developed to use SCTP
  - SIP over SCTP
  - HTTP over SCTP
  - recommended transport protocol for DIAMETER

- Strong security can be provided via TLS [35]

Maguire
maguire@it.kth.se
Stream Control Transmission Protocol (SCTP) [26]
2006.03.13
SCTP 301 of 342
Internetworking/Internetteknik

SCTP Header

0 7 8 15 16 23 24 31	
16 bit source port	16 bit destination port
32 bit verification tag	
32 bit Checksum	
Control chunk(s) {If any}	
Data chunk(s) {If any}	

(12 bytes)
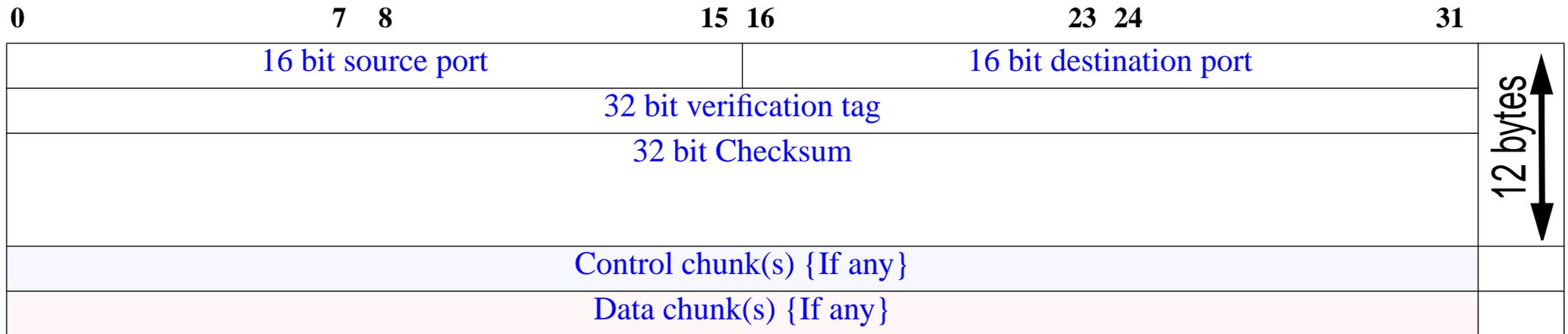
Figure 38: SCTP packet (see Forouzan figure 13.4 pg. 350)

IP protocol x84 = SCTP

- ## General Header
  - As with UDP & TCP, SCTP provides de/multiplexing via the 16 bit source and destination ports.
  - Associations between end points are defined by a unique **verification tag**
    - A separate verification tag is used in each direction
  - SCTP applies a CRC-32 end-to-end checksum to its general header and all the chunks
    - Previously it used Adler-32 checksum [34]
- ## Chunks
  - Control information is contained in Control Chunks (these always **precede** any data chunks)
  - Multiple data chunks can be present - each containing data for different streams

Maguire
maguire@it.kth.se

SCTP Header
2006.03.13

SCTP 302 of 342
Internetworking/Internetteknik

SCTP Chunk

0	7	8	15	16	23	24	31
Type		Flag		Length			
Chunk information padded to a multiple of 4 bytes							
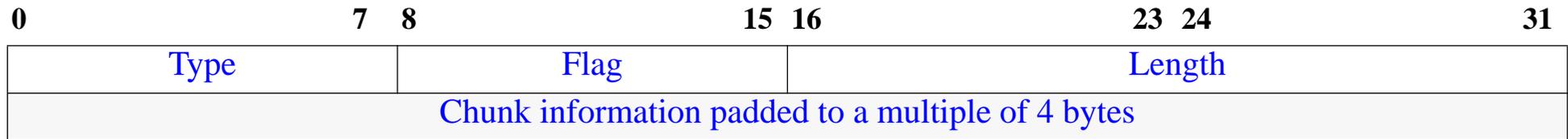
Figure 39: SCTP packet (see Forouzan figure 13.8 pg. 354)

- Type

Type	Chunk	Description	Type	Chunk	Description
0	DATA	User data	7	SHUTDOWN	Terminate an association
1	INIT	Setup an association	8	SHUTDOWN-ACK	Acknowledge SHUTDOWN chunk
2	INIT-ACK	Acknowledge an INIT chunk	9	ERROR	Reports errors without shutting down
3	SACK	Selective Acknowledgement	10	COOKIE ECHO	Third packet in establishment of an association
4	HEARTBEAT	Probe to see if peer is alive	11	COOKIE ACK	Acknowledges COOKIE ECHO chunk
5	HEARTBEAT-ACK	Acknowledgement of a HEARTBEAT chunk	14	SHUTDOWN COMPLETE	Third packet in an association terminations
6	ABORT	Abort an association	192	FORWARD TSN	To adjust the cumulative TSN

- Flag - 8 bit field defined per chunk type
- Length - 16 bit length of chunk including chunk header (i.e., smallest value is 4) - does **not** include any padding bytes (hence you know just how much padding there is)

Maguire
maguire@it.kth.se
SCTP Chunk
2006.03.13
SCTP 303 of 342
Internetworking/Internetteknik

Association establishment - 4-way handshake
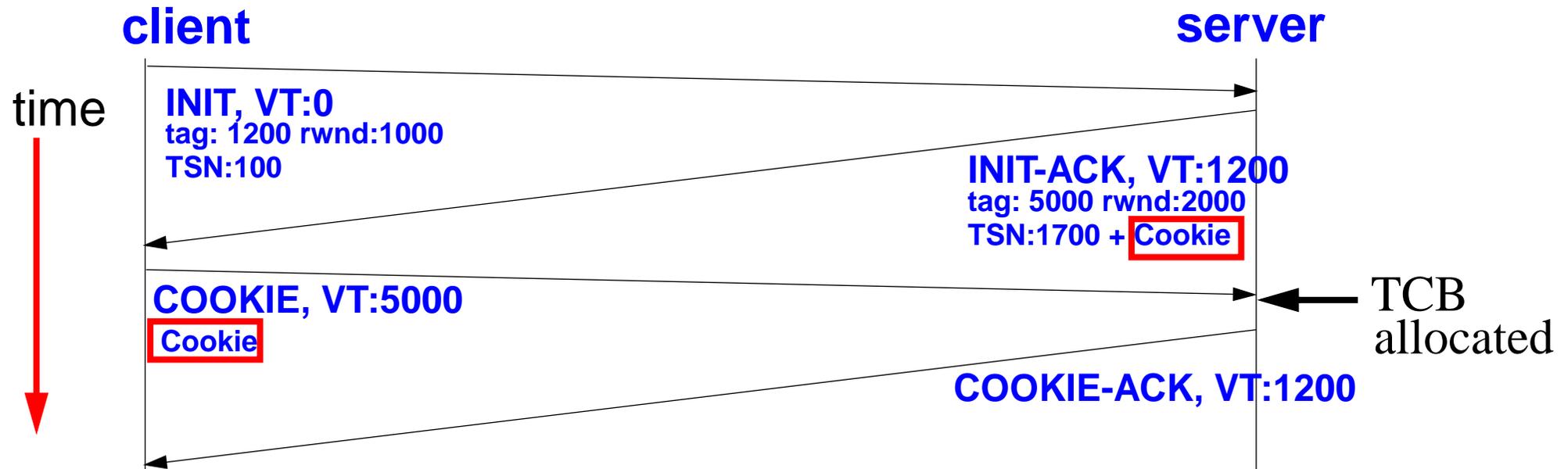
See Forouzan figure 13.19 page 363

The entity initiating the connection is (normally) called the "client" and does an active open, where as the server needed to previous do a passive open.

Maguire
maguire@it.kth.se

Association establishment - 4-way handshake
2006.03.13

SCTP 304 of 342
Internetworking/Internetteknik

INIT Chunk

0 7	8 15	16 23 24 31
Type = 1	Flag = 0	Length
Initiation tag		
Advertised receiver window credit (rwnd)		
Outbound streams		Maximum inbound streams
Initial Transmission sequence number (TSN)		
variable-length parameters (optional)		

Figure 40: SCTP INIT chunk (see Forouzan figure 13.10 pg. 357)

- **Initiation tag**
  - defines the tags for this association to be used by the other party
  - reduce the risk due to a blind attacker (since there is only a 1 in 2^{32} chance of guessing the right tag)
  - can reject delayed packets - thus avoiding the need for TCP's TIME-WAIT timer
- **Advertised receiver window credit**
  - defines rwnd (i.e., how much the receiver can send to this party)
- **Outbound streams**
  - suggested upper number of streams **from** this sender (can be reduced by receiver)
- Maximum inbound streams
  - upper limit of streams **to** this sender

Maguire
maguire@it.kth.se

INIT Chunk
2006.03.13

SCTP 305 of 342
Internetworking/Internetteknik

- **Transmission sequence number (TSN)**
  - Initializes the TSN in the outbound direction, initialized to a random value
- **Variable-length parameters**
  - IP address(es) of endpoint
    - Multiple addresses are used to support multihoming
    - The **receiver** selects the primary address for the other endpoint
  - Type of addresses
  - Support for Explicit Congestion Notification (ECN)
  - …

Maguire
maguire@it.kth.se

INIT Chunk
2006.03.13

SCTP 306 of 342
Internetworking/Internetteknik

INIT ACK Chunk

Type = 2	Flag = 0	Length
Initiation tag		
Advertised receiver window credit (rwnd)		
Outbound streams		Maximum inbound streams
Initial Transmission sequence number (TSN)		
Parameter type: 7		Parameter length
State Cookie		
variable-length parameters (optional)		

Column bit positions: 0, 7 8, 15 16, 23 24, 31
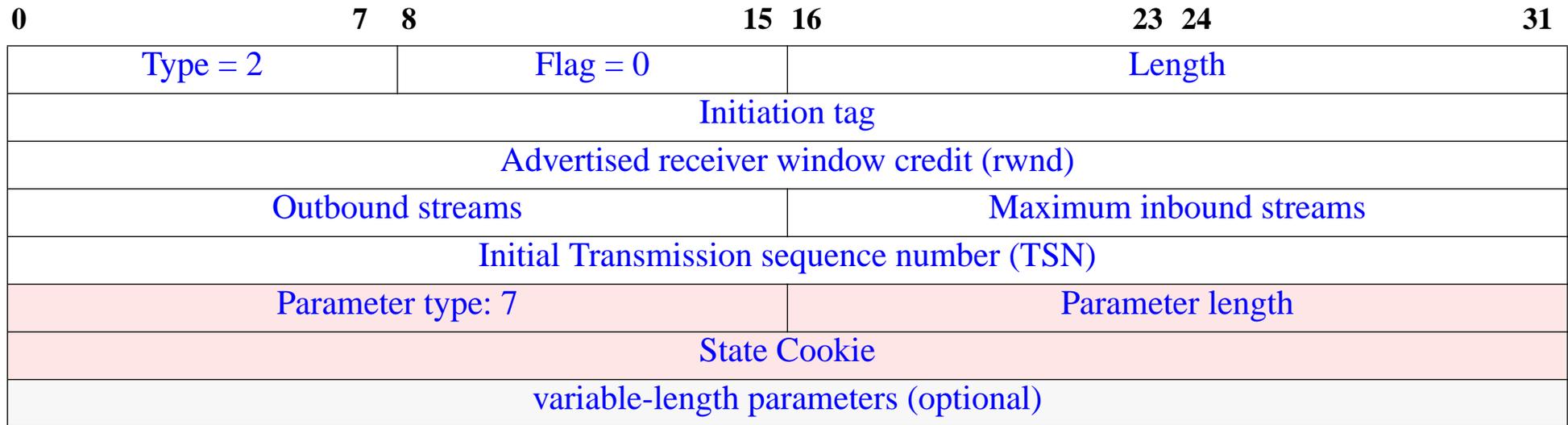
Figure 41: SCTP INIT ACK chunk (see Forouzan figure 13.11 pg. 358)

The same fields as in the INIT chunk (with Initiation tag value set to that of the INIT) - but with the addition of a required parameter with a state cookie.

- Parameter type: 7 = State Cookie
- Parameter length = size of State Cookie + 4 (the parameter type and length fields)

A packet carrying this INIT ACK chunk can not contain any other control or data chunks.

Maguire
maguire@it.kth.se

INIT ACK Chunk
2006.03.13

SCTP 307 of 342
Internetworking/Internetteknik

State Cookie

Use of the COOKIE prevents a SYN flood like attack - since resources are not allocated until the COOKIE ECHO chuck is received.

However, state has to be saved from the initial INIT chunk - therefore it is placed in the cookie in a way that only the server can access it (hence the cookie is sealed with an HMAC {aka digest} after being created {aka "baked"}). This requires that the server has a secret key which it uses to compute this digest.

If the sender of the INIT is an attacker located on another machine, they won't be able to receive the cookie if they faked the source address in the INIT - since the INIT ACK is sent to the address and contains the cookie!

- Without a cookie $\Rightarrow$ no association is created and no resources (such as TCB) are tied up!

Maguire
maguire@it.kth.se
INIT ACK Chunk
2006.03.13
SCTP 308 of 342
Internetworking/Internetteknik

COOKIE ECHO Chunk

0	7	8	15	16	23	24	31
Type = 10		Flag = 0		Length			
State Cookie							
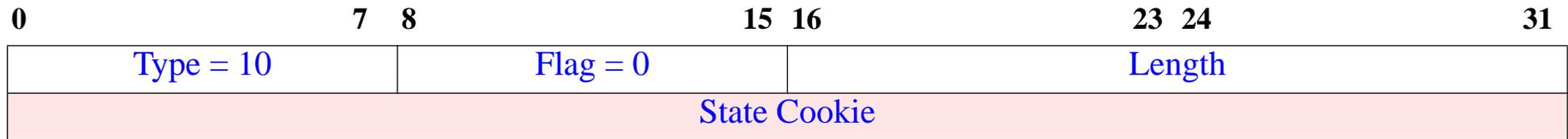
Figure 42: SCTP COOKIE ECHO chunk (see Forouzan figure 13.12 pg. 359)

- (chunk) Type: 10 = COOKIE ECHO
- (chunk) length = size of State Cookie + 4 (the parameter type and length fields)
- State Cookie
  - simply a copy of the COOKIE data from the INIT ACK chunk
  - The COOKIE data is opaque (i.e., only the sender can read the cookie)

A packet carrying this COOKIE ECHO chunk can contain other control or data chunks -- in particular it can care the first user (client) data!

Maguire
maguire@it.kth.se

COOKIE ECHO Chunk
2006.03.13

SCTP 309 of 342
Internetworking/Internetteknik

COOKIE ACK Chunk

0	7	8	15 16	23 24	31

Type = 11	Flag = 0	Length = 4
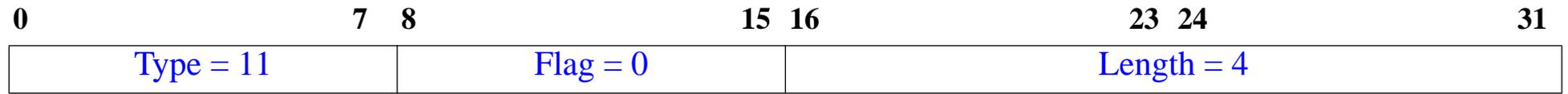
Figure 43: SCTP COOKIE ACK chunk (see Forouzan figure 13.13 pg. 359)

Completes the 4 way handshake.

A packet with this chunk can also carry control and data chunks (in particular the first of the user (server) data.

Maguire
maguire@it.kth.se

COOKIE ACK Chunk
2006.03.13

SCTP 310 of 342
Internetworking/Internetteknik

Data Chunk

0	7	8	15	16	23	24	31
Type = 0		Reserved	U	B	E	Length		
Transmission sequence number (TSN)							
Stream identifier (SI)				Stream Sequence number (SSN)			
Protocol Identifier							
User data							
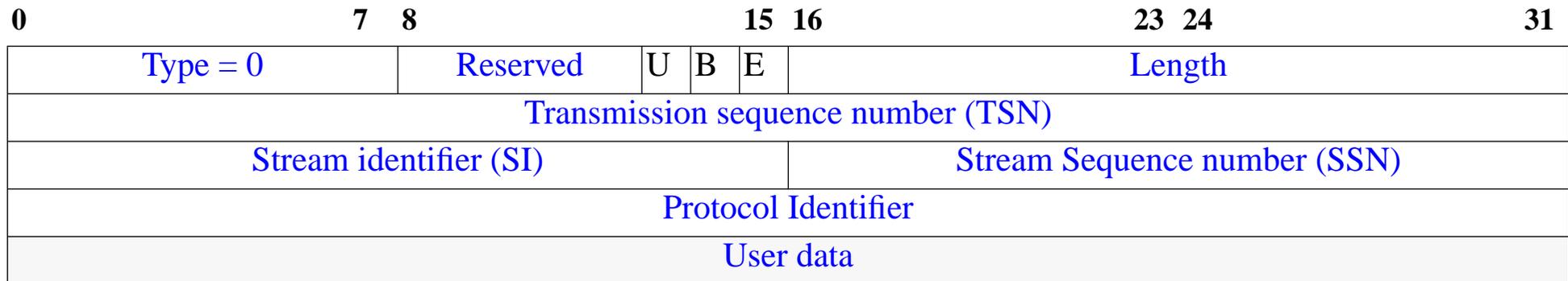
Figure 44: SCTP Data Chunk (see Forouzan figure 13.9 pg. 356)

- **Flags:**
  - U - Unordered - for delivery to the application right away
  - B - Beginning (chunk position - for use with fragmentation)
  - E - End chunk

- **Transmission sequence number (TSN)**- only data chunks consume TSNs

- **Stream identifier (SI)**

- **Stream Sequence number (SSN)**

- **Protocol Identifier**

- **User data**

  - at least 1 byte of user data; padded to 32 bit boundaries
  - although a message can be spread over multiple chunks, each chunk contains data from **only** a single message (like UDP, each message results in one or more data SCTP chunks)

Maguire
maguire@it.kth.se

Data Chunk
2006.03.13

SCTP 311 of 342
Internetworking/Internetteknik
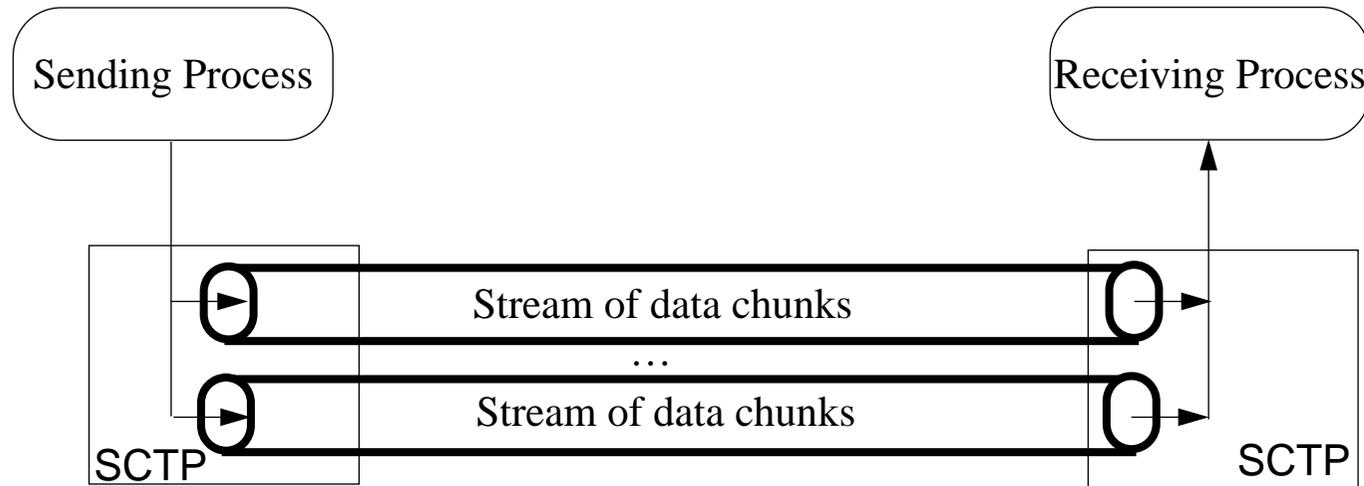
Multiple-Streams

Figure 45: Multiple-streams (see Forouzan figure 13.2 pg. 347)

The figure above shows a **single** association.

Each stream has a unique stream identifier (SI) and maintains its own stream sequence number (SSN).

Unordered data chunks (i.e., with $U = 0$) - do **not** consume a SSN and are delivered when they arrive at the destination.

Multiple streams and unordered data avoid TCP's head of line blocking.

Maguire
maguire@it.kth.se

Multiple-Streams
2006.03.13

SCTP 312 of 342
Internetworking/Internetteknik

Selective Acknowledgement (SACK) Chunk

0 7	8 15	16 23 24 31
Type = 3	Flag = 0	Length
cumulative TSN acknowledgement		
Advertised receiver window credit		
Number of gap ACK blocks: N		Number of duplicates: M
Gap ACK block #1 start TSN offset		Gap ACK block #1 end TSN offset
…		…
Gap ACK block #N start TSN offset		Gap ACK block #N end TSN offset
Duplicate TSN 1		
…		
Duplicate TSN M		

Figure 46: SCTP Data Chunk (see Forouzan figure 13.9 pg. 356)

- Cumulative Transmission sequence number (TSN) acknowledgement - the last data chunk received **in sequence**

- Gap = received sequence of chunks (indicated with start .. end TSNs)

- Duplicate TSN - indicating duplicate chunks (if any)

- SACK always sent to the IP address where the corresponding packet originated

Maguire
maguire@it.kth.se

Selective Acknowledgement (SACK) Chunk
2006.03.13

SCTP 313 of 342
Internetworking/Internetteknik

ERROR chunk

Sent when an endpoint finds some error in a packet

0	7	8	15	16	23	24	31
Type = 9		Flag		Length			
One or more causes (optional)							
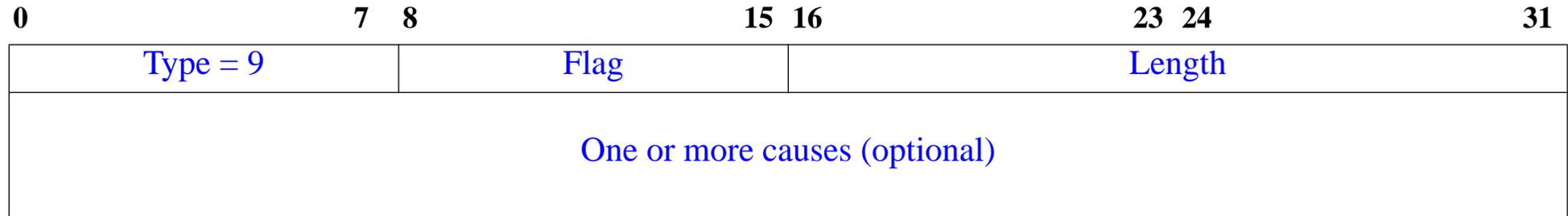
Figure 47: SCTP ERROR chunk (see Forouzan figure 13.17 pg. 361)

Error code	Description
1	Invalid Stream identifier
2	Missing mandatory parameter
3	State cookie error
4	Out of resource
5	Unresolvable address
6	Unrecognized chunk type
7	Invalid mandatory parameters
8	Unrecognized parameter
9	No user data
10	Cookie received while shutting down

Maguire
maguire@it.kth.se

ERROR chunk
2006.03.13

SCTP 314 of 342
Internetworking/Internetteknik

Association Termination

Two forms of termination

- ## Association Abort
  - Used in the event of a fatal error
  - uses same error codes as the ERROR Chunk
  - Chunk format

0	7	8	15	16	23	24	31
Type = 6		Flag = 6		Length			
One or more causes (optional)							
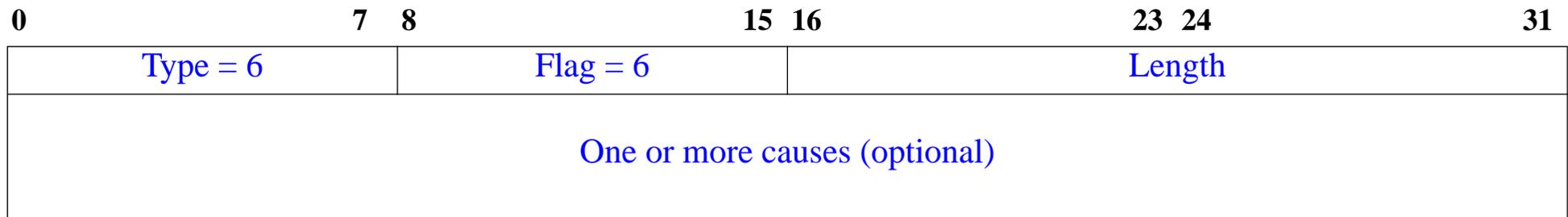
Figure 48: SCTP ABORT chunk (see Forouzan figure 13.18 pg. 362)

- ## Association Shutdown - graceful termination

Maguire
maguire@it.kth.se

Association Termination
2006.03.13

SCTP 315 of 342
Internetworking/Internetteknik

Association Shutdown

Figure 49: Adapted from Forouzan figure 13.21 page 368 and slide 15 of [25]

0		7	8		15	16		23	24		31
Type = 7			Flag			Length = 8					
cumulative TSN acknowledgement											
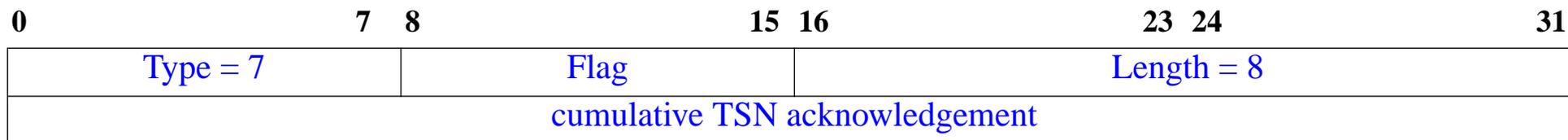
Figure 50: SCTP SHUTDOWN chunk (see Forouzan figure 13.16 pg. 361)

0		7	8		15	16		23	24		31
Type = 8			Flag			Length = 4					
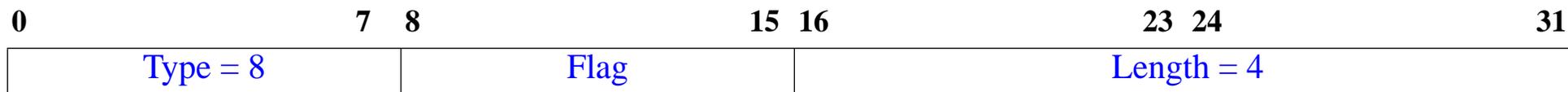
Figure 51: SCTP SHUTDOWN ACK chunk (see Forouzan figure 13.16 pg. 361)

Maguire
maguire@it.kth.se

Association Termination
2006.03.13

SCTP 316 of 342
Internetworking/Internetteknik

0	7	8	15	16	23	24	31
Type = 14		Flag		T	Length = 4		
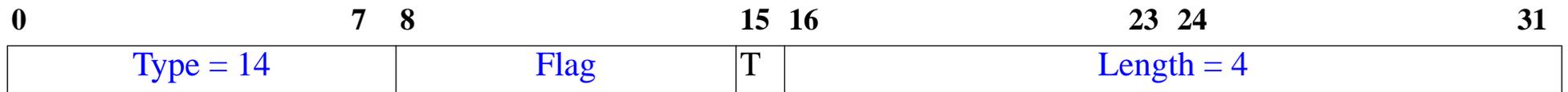
Figure 52: SCTP SHUTDOWN COMPLETE chunk (see Forouzan figure 13.16 pg. 361)

- T bit indicates the sender did **not** have a Transmission Control Block (TCB)

Maguire
maguire@it.kth.se
Association Termination
2006.03.13
SCTP 317 of 342
Internetworking/Internetteknik

SCTP Example - Daytime [29]

```
server# ./daytime_server -s 192.168.1.2 -vv
1        : Communication up (1 paths)
1        : Network status change: path 0 is now REACHABLE
1        : Shutdown complete

client# ./terminal -vv -r 13 -d 192.168.1.2 -s 192.168.1.1
1        : Communication up (1 paths, 1 In-Streams, 1 Out-Streams)
1        : Network status change: path 0 (towards 192.168.1.2) is now REACHABLE
Wed Apr 27 11:52:04 2005
1        : Shutdown received
```

```
11 74.864232 192.168.1.1        192.168.1.2        SCTP        INIT
12 74.864552 192.168.1.2        192.168.1.1        SCTP        INIT_ACK
13 74.864808 192.168.1.1        192.168.1.2        SCTP        COOKIE_ECHO
14 74.865073 192.168.1.2        192.168.1.1        SCTP        COOKIE_ACK
15 74.865273 192.168.1.2        192.168.1.1        SCTP        DATA
16 74.865733 192.168.1.1        192.168.1.2        SCTP        SACK
17 74.865933 192.168.1.2        192.168.1.1        SCTP        SHUTDOWN
18 74.866132 192.168.1.1        192.168.1.2        SCTP        SHUTDOWN_ACK
19 74.866195 192.168.1.2        192.168.1.1        SCTP        SHUTDOWN_COMPLETE
```

Figure 53: SCTP Daytime example - output from Ethereal

Maguire
maguire@it.kth.se

SCTP Example - Daytime [29]
2006.03.13

SCTP 318 of 342
Internetworking/Internetteknik

ethereal capture - daytime - INIT

```
Frame 11 …

Stream Control Transmission Protocol
     Source port: 10777
     Destination port: 13
     Verification tag: 0x00000000

     Checksum: 0x2b84fdb0¹
     INIT chunk (Outbound streams: 10, inbound streams: 10)
          Chunk type: INIT (1)
          Chunk flags: 0x00
          Chunk length: 32
          Initiate tag: 0x43d82c5d
          Advertised receiver window credit (a_rwnd): 131071
          Number of outbound streams: 10
          Number of inbound streams: 10
          Initial TSN: 771212194
          Forward TSN supported parameter
               Parameter type: Forward TSN supported (0xc000)
  …            Parameter length: 4
          Supported address types parameter (Supported types: IPv4)
               Parameter type: Supported address types (0x000c)
               Parameter length: 6
               Supported address type: IPv4 address (5)
```

---

1. Ethereal complains about this checksum saying "(incorrect Adler32, should be 0x973b078d)", but this is in error see [34].

Maguire
maguire@it.kth.se
ethereal capture - daytime - INIT
2006.03.13
SCTP 319 of 342
Internetworking/Internetteknik

ethereal capture - daytime - INIT-ACK

```
Frame 12 … Stream Control Transmission Protocol
      Source port: 13
      Destination port: 10777
      Verification tag: 0x43d82c5d
      Checksum: 0x7f61f237
      INIT_ACK chunk (Outbound streams: 1, inbound streams: 1)
            Chunk type: INIT_ACK (2)
            Chunk flags: 0x00
            Chunk length: 128
            Initiate tag: 0x5d581d9a
            Advertised receiver window credit (a_rwnd): 131071
            Number of outbound streams: 1
            Number of inbound streams: 1
            Initial TSN: 1514529259
            State cookie parameter (Cookie length: 100 bytes)
                  Parameter type: State cookie (0x0007)
                     Parameter length: 104
                  State cookie: 5D581D9A0001FFFF000100015A45E1EB...
            Forward TSN supported parameter
                  Parameter type: Forward TSN supported (0xc000)
                        1... .... .... .... = Bit: Skip parameter and con-
   tinue prosessing of the chunk
                        .1.. .... .... .... = Bit: Do report
                  Parameter length: 4
```

Maguire
maguire@it.kth.se

ethereal capture - daytime - INIT-ACK
2006.03.13

SCTP 320 of 342
Internetworking/Internetteknik

ethereal capture - daytime - COOKIE-ECHO

Frame 13 …

```
        Source port: 10777
        Destination port: 13
        Verification tag: 0x5d581d9a
        Checksum: 0x3af3f579
        COOKIE_ECHO chunk (Cookie length: 100 bytes)
            Chunk type: COOKIE_ECHO (10)
                0... .... = Bit: Stop processing of the packet
                .0.. .... = Bit: Do not report
            Chunk flags: 0x00
            Chunk length: 104
            Cookie: 5D581D9A0001FFFF000100015A45E1EB...
```

ethereal capture - daytime - COOKIE-ACK

Frame 14 …

```
     Source port: 13
     Destination port: 10777
     Verification tag: 0x43d82c5d
     Checksum: 0x762d80d7
     COOKIE_ACK chunk
          Chunk type: COOKIE_ACK (11)
          Chunk flags: 0x00
          Chunk length: 4
```

Maguire
maguire@it.kth.se

ethereal capture - daytime - COOKIE-ACK
2006.03.13

SCTP 322 of 342
Internetworking/Internetteknik

ethereal capture - daytime - DATA

Frame 15 …

```
      Source port: 13
      Destination port: 10777
      Verification tag: 0x43d82c5d
      Checksum: 0xf8fb1754
      DATA chunk(ordered, complete segment, TSN: 1514529259, SID: 0,
SSN: 0, PPID: 0, payload length: 25 bytes)
          Chunk type: DATA (0)
          Chunk flags: 0x03
              .... ...1 = E-Bit: Last segment
              .... ..1. = B-Bit: First segment
              .... .0.. = U-Bit: Ordered deliviery
          Chunk length: 41
          TSN: 1514529259
          Stream Identifier: 0x0000
          Stream sequence number: 0
          Payload protocol identifier: not specified (0)
          Chunk padding: 000000
Data (25 bytes)
0000   57 65 64 20 41 70 72 20 32 37 20 31 31 3a 34 33    Wed Apr 27 11:43
0010   3a 32 32 20 32 30 30 35 0a                         :22 2005.
```

Maguire
maguire@it.kth.se

ethereal capture - daytime - DATA
2006.03.13

SCTP 323 of 342
Internetworking/Internetteknik

ethereal capture - daytime - SACK

Frame 16 …
```
      Source port: 10777
      Destination port: 13
      Verification tag: 0x5d581d9a
      Checksum: 0xfa994e35
    SACK chunk (Cumulative TSN: 1514529259, a_rwnd: 131071, gaps: 0,
duplicate TSNs: 0)
          Chunk type: SACK (3)
          Chunk flags: 0x00
          Chunk length: 16
          Cumulative TSN ACK: 1514529259
          Advertised receiver window credit (a_rwnd): 131071
          Number of gap acknowldgement blocks : 0
          Number of duplicated TSNs: 0
```

Maguire
maguire@it.kth.se

ethereal capture - daytime - SACK
2006.03.13

SCTP 324 of 342
Internetworking/Internetteknik

ethereal capture - daytime - SHUTDOWN

Frame 17 …

```
    Source port: 13
    Destination port: 10777
    Verification tag: 0x43d82c5d
    Checksum: 0xf447d00f
    SHUTDOWN chunk (Cumulative TSN ack: 771212193)
        Chunk type: SHUTDOWN (7)
        Chunk flags: 0x00
        Chunk length: 8
        Cumulative TSN Ack: 771212193
```

Maguire
maguire@it.kth.se

ethereal capture - daytime - SHUTDOWN
2006.03.13

SCTP 325 of 342
Internetworking/Internetteknik

ethereal capture - daytime - SHUTDOWN_ACK

Frame 18

```
    Source port: 10777
    Destination port: 13
    Verification tag: 0x5d581d9a
    Checksum: 0x9f44d056
    SHUTDOWN_ACK chunk
        Chunk type: SHUTDOWN_ACK (8)
        Chunk flags: 0x00
        Chunk length: 4
```

Maguire
maguire@it.kth.se

ethereal capture - daytime - SHUTDOWN_ACK
2006.03.13

SCTP 326 of 342
Internetworking/Internetteknik

ethereal capture - daytime - SHUTDOWN_COMPLETE

Frame 19…

```
    Source port: 13
    Destination port: 10777
    Verification tag: 0x43d82c5d
    Checksum: 0x3db6e771
    SHUTDOWN_COMPLETE chunk
        Chunk type: SHUTDOWN_COMPLETE (14)
        Chunk flags: 0x00
            .... ...0 = T-Bit: TCB destroyed
        Chunk length: 4
```

Maguire
maguire@it.kth.se

ethereal capture - daytime - SHUTDOWN_COMPLETE
2006.03.13

SCTP 327 of 342
Internetworking/Internetteknik

Fault Management

- Endpoint Failure Detection
  - Endpoint keeps a counter of the total number of consecutive retransmissions to its peer (including retransmissions to all the destination transport addresses [= port + IP address] of the peer if it is multi-homed). When this counter exceeds 'Association.Max.Retrans', the endpoint will consider the peer endpoint unreachable and shall stop transmitting any more data to it (the association enters the CLOSED state).
  - Counter is reset each time:
    - a DATA chunk sent to that peer is acknowledged (by the reception of a SACK) or
    - a HEARTBEAT-ACK is received from the peer

- Path Failure Detection
  - Each time (1) T3-rtx timer expires on any address or (2) a HEARTBEAT sent to an idle address is not acknowledged within a RTO, then the error counter of that destination will be incremented. When this error counter exceeds 'Path.Max.Retrans' for that destination address, then the endpoint marks the destination transport address as inactive and notifies the upper layer.
  - the endpoint clears the error counter of this destination transport address when:
    - an outstanding TSN is acknowledged or
    - a HEARTBEAT address is acknowledged
  - When the primary path is marked inactive, then the sender **may** automatically transmit new packets to an alternate destination address if one exists and is active
    - If more than one alternate address is active $\Rightarrow$ only **one** transport address is chosen as the new destination transport address.

Maguire
maguire@it.kth.se

Fault Management
2006.03.13

SCTP 328 of 342
Internetworking/Internetteknik

HEARTBEAT and HEARTBEAT ACK Chunks

0	7 8	15 16	23 24	31
Type = 4 or 5		Flag = 0	Length	
Parameter type: 1			Parameter length	
Sender specific information				
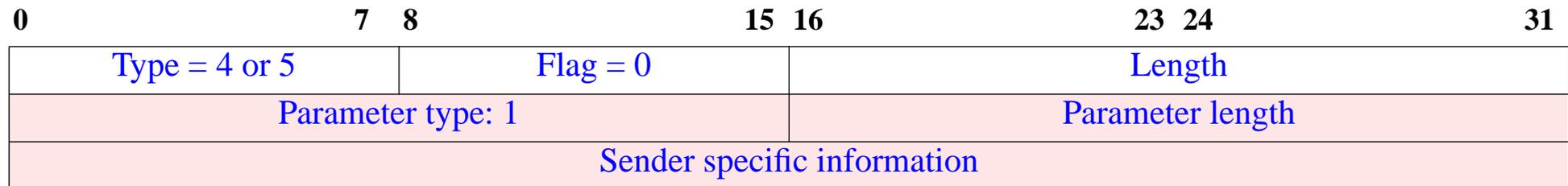
Figure 54: SCTP HEARTBEAT and HEARTBEAK ACK chunks (see Forouzan figure 13.15 pg. 360)

- (chunk) Type: 4 = HEARTBEAT
- (chunk) Type: 5 = HEARTBEAT ACK
- (chunk) length = size of sender specific information + 4 (the parameter type and length fields)
- Sender specific information
  - The sender puts its Local time and transport address in (note that the sctplib implementation 1.0.2 puts the time in as an unsigned 32 bit integer and puts the path index in (also as an unsigned 32 bit integer) and add a HMAC computed over these values [29]
  - The acknowledgement simply contains a copy of this information

Heartbeats every ~30 seconds.

Maguire
maguire@it.kth.se

HEARTBEAT and HEARTBEAT ACK Chunks
2006.03.13

SCTP 329 of 342
Internetworking/Internetteknik

Heartbeat and ACK

```
Frame x …
    Source port: 9
    Destination port: 38763
    Verification tag: 0x36fab554
      Checksum: 0x0e6c8d88 (incorrect Adler32, should be 0xf5340ec5)
    HEARTBEAT chunk (Information: 28 bytes)
        Chunk type: HEARTBEAT (4)
        Chunk flags: 0x00
        Chunk length: 32
        Heartbeat info parameter (Information: 24 bytes)
            Parameter type: Heartbeat info (0x0001)
            Parameter length: 28
            Heartbeat information: 0280351E00000000E1A06CFBC1C6933F...
    Source port: 38763
    Destination port: 9
    Verification tag: 0x57c3a50c

        Checksum: 0xaa2fba80 (incorrect Adler32, should be 0xe7450e58)

    HEARTBEAT_ACK chunk (Information: 28 bytes)
        Chunk type: HEARTBEAT_ACK (5)
        Chunk flags: 0x00
        Chunk length: 32
        Heartbeat info parameter (Information: 24 bytes)
            Parameter type: Heartbeat info (0x0001)
            Parameter length: 28
            Heartbeat information: 0280351E00000000E1A06CFBC1C6933F...
```

Maguire
maguire@it.kth.se

Heartbeat and ACK
2006.03.13

SCTP 330 of 342
Internetworking/Internetteknik

Differences from TCP Congestion Control

- Any DATA chunk that has been acknowledged by SACK, including DATA that arrived out of order, are **only** considered fully delivered when the Cumulative TSN Ack Point passes the TSN of the DATA chunk

$\Rightarrow$ cwnd controls the amount of outstanding data, rather than (as in the case of non-SACK TCP) the upper bound between the highest acknowledged sequence number and the latest DATA chunk that can be sent within the congestion window

$\Rightarrow$ different fast-retransmit & fast-recovery than non-SACK TCP

- Retransmission based on both retransmission timer (with an RTO per path)
- Three SACKS (i.e., 4 consecutive duplicate SACKs indicating missing chunks) fi immediate retransmission of these missing chunks

Sender

- uses the same destination address until instructed by the upper layer (however, SCTP may change to an alternate destination in the event an address is marked inactive) $\Rightarrow$ retransmission can be to a different transport address than the original transmission.
- keeps separate congestion control parameters (cwnd, ssthresh, and partial_bytes_acked) for each of the destination addresses it can send to (i.e., not each source-destination pair)
  - these parameters should decay if the address is not used
  - does slow-start upon the first transmission to each of destination addresses

Maguire
maguire@it.kth.se

Differences from TCP Congestion Control
2006.03.13

SCTP 331 of 342
Internetworking/Internetteknik

Path MTU Discovery

- IPv4
  - Based on RFC 1191 [31] each endpoint maintains an estimate of the maximum transmission unit (MTU) along a **each** path and refrains from sending packets along that path which exceed the MTU, other than occasional attempts to probe for a change in the Path MTU (PMTU).

- IPv6
  - Based on RFC1981 [32] an SCTP sender using IPv6 **must** use Path MTU Discovery, unless all packets are less than the minimum IPv6 MTU (see RFC 2460 [33]).

SCTP differs in several ways from the description in RFC 1191 of applying MTU discovery to TCP:

1 SCTP associations can span multiple addresses $\Rightarrow$ an endpoint does PMTU discovery on a per-destination-address basis
  - The term "MTU" always refers to the MTU associated with the destination address

2 Since SCTP does not have a notion of "Maximum Segment Size", for each destination $MTU_{initial} \leq MTU_{link}$ for the local interface to which packets for that remote destination address will be routed

Maguire
maguire@it.kth.se

Path MTU Discovery
2006.03.13

SCTP 332 of 342
Internetworking/Internetteknik

3 When retransmitting to a remote address for which the IP datagram appears too large for the path MTU to that address, the IP datagram **should** be retransmitted without the DF bit set, enabling it to be fragmented. While *initial* transmissions of IP datagrams **must** have DF set.

4 Sender maintains an association PMTU (= smallest PMTU discovered for all of the peer's destination addresses); when fragmenting messages this association PMTU is used to calculate the size of each fragment ⇒ retransmissions can sent to an alternate address without encountering IP fragmentation

Maguire
maguire@it.kth.se

Path MTU Discovery
2006.03.13

SCTP 333 of 342
Internetworking/Internetteknik

SCTP header continued

- **Reliability** is provided by a 32 bit SCTP sequence numbers (TSN)
  - The initial sequence number is a random 32 bit number
  - These sequence numbers are in the header of individual chunks
  - This cumulative number is used to provide both flow control and error control
- SCTP resequences data at the receiving side
- SCTP discards duplicate data at the receiving side

The window size (or more exactly the receive window size (rwnd)) - indicates how many bytes the receiver is prepared to receive (this number is relative to the acknowledgement number).

Maguire
maguire@it.kth.se

SCTP header continued
2006.03.13

SCTP 334 of 342
Internetworking/Internetteknik

Forward Cumulative TSN

Allows an endpoint to signal to its peer that it should move the cumulative acknowledgement forward [30]. This protocol extension adds a new parameter (Forward-TSN-Supported) to INIT and INIT ACK, and a new FORWARD TSN chunk type. It provides an example of a partially reliable service.

0	7	8	15	16	23	24	31
Type = 192		Flag = 0		Length			
New cumulative TSN							
Stream #1				Stream Sequence #1			
…				…			
Stream #N				Stream Sequence #N			

Figure 55: SCTP FORWARD TSN Chunk (see [30])

- Stream$_i$ a stream number that was skipped by this FWD-TSN.
- Stream Sequence$_i$ = the largest stream sequence number in stream$_i$ being skipped

Maguire
maguire@it.kth.se

Forward Cumulative TSN
2006.03.13

SCTP 335 of 342
Internetworking/Internetteknik

- Receiver can use the $Stream_i$ and Stream $Sequence_i$ fields to enable delivery of (stranded) TSN's that remain in the stream re-ordering queues.

Maguire
maguire@it.kth.se

Forward Cumulative TSN
2006.03.13

SCTP 336 of 342
Internetworking/Internetteknik

SCTP Performance

See the upcoming exjobb report by Mia Immonen.

Maguire

maguire@it.kth.se

Forward Cumulative TSN

2006.03.13

SCTP 337 of 342

Internetworking/Internetteknik

Transport Protocol Functional Overview

From table 1-1 of [27] on page 12; also appears in [36]

Protocol Feature	SCTP	TCP	UDP
State required at each endpoint	Yes	Yes	No
Reliable data transfer	Yes	Yes	No
Congest control and avoidance	Yes	Yes	No
Message boundary conservation	Yes	No	Yes
Path MTU discovery and message fragmentation	Yes	Yes	No
Message bundling	Yes	Yes	No
Multi-homed hosts support	Yes	No	No
Multi-stream support	Yes	No	No
Unordered data delivery	Yes	No	Yes
Security cookie against SYN flood attack	Yes	No	No
Built-in heartbeat (readability check)	Yes	No	No

Maguire
maguire@it.kth.se

Transport Protocol Functional Overview
2006.03.13

SCTP 338 of 342
Internetworking/Internetteknik

Summary

This lecture we have discussed:

- ## SCTP
  - Message framing
  - Multi-homing
  - Multi-streaming

- ## How SCTP differs from TCP

- ## Measurements of an implementation (there are other implementations such as that included with [27]):
  - *http://www.sctp.de*
  - *http://www.sctp.org*
  - Linux Kernel SCTP *http://sourceforge.net/projects/lksctp*

Maguire
maguire@it.kth.se

Summary
2006.03.13

SCTP 339 of 342
Internetworking/Internetteknik

References

[24] G. Sidebottom, K. Morneault, and J. Pastor-Balbas, "Signaling System 7 (SS7) Message Transfer Part 3 (MTP3) - User Adaptation Layer (M3UA)", IETF RFC 3332, September 2002 *http://www.ietf.org/rfc/rfc3332.txt*

[25] Andreas Jungmaier, "A Gentle Introduction to SCTP", 19th Chaos Communications Congress, Berlin, 2002

http://tdrwww.exp-math.uni-essen.de/inhalt/forschung/19ccc2002/html/slide_1.html

[26] R. Stewart, Q. Xie, K. Morneault, C. Sharp, H. Schwarzbauer, T. Taylor, I. Rytina, M. Kalla, L. Zhang, and V. Paxson, "Stream Control Transmission Protocol", IETF RFC 2960, October 2000 *http://www.ietf.org/rfc/rfc2960.txt*

[27] Randall R. Stewart and Qiaobing Xie, "Stream Control Transmission Protocol: A Reference Guide", Addison-Wesley, 2002, ISBN 0-201-72186-4.

[28] K. Morneault, S. Rengasami, M. Kalla, and G. Sidebottom, "ISDN

Maguire
maguire@it.kth.se

References
2006.03.13

SCTP 340 of 342
Internetworking/Internetteknik

Q.921-User Adaptation Layer", IETF RFC 3057, February 2001

http://www.ietf.org/rfc/rfc3057.txt

[29] Andreas Jungmaier , Herbert Hölzlwimmer, Michael Tüxen , and Thomas Dreibholz, "sctplib-1.0.2", Siemens AG and the Institute of Computer Networking Technology, University of Essen, Germany, August 2004 *http://www.sctp.de/sctp-download.html* {Note that a later version 1.0.3 was released March 4th, 2005}

[30] R. Stewart, M. Ramalho, Q. Xie, M. Tuexen, and P. Conrad, "Stream Control Transmission Protocol (SCTP) Partial Reliability Extension", IETF RFC 3758, May 2004 *http://www.ietf.org/rfc/rfc3758.txt*

[31] J. Mogul and S. Deering, "Path MTU Discovery", IETF RFC 1191, November 1990 *http://www.ietf.org/rfc/rfc1191.txt*

[32] J. McCann, S. Deering, and J. Mogul, "Path MTU Discovery for IP version 6", IETF RFC 1981, August 1996 *http://www.ietf.org/rfc/rfc1981.txt*

[33] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6)

Maguire
maguire@it.kth.se
References
2006.03.13
SCTP 341 of 342
Internetworking/Internetteknik

Specification", IETF RFC 2460, December 1998 *http://www.ietf.org/rfc/rfc2460.txt*

[34] J. Stone, R. Stewart, and D. Otis, "Stream Control Transmission Protocol (SCTP) Checksum Change", IETF RFC 3309, September 2002

http://www.ietf.org/rfc/rfc3309.txt

[35] A. Jungmaier, E. Rescorla, and M. Tuexen, "Transport Layer Security over Stream Control Transmission Protocol", IETF RFC 3436, December 2002

http://www.ietf.org/rfc/rfc3436.txt

[36] "SCTP Primer", Mon, Mar 1, 2004 03:35:54 PM

http://datatag.web.cern.ch/datatag/WP3/sctp/primer.htm

[37] Mia Immonen, "SIGTRAN: Signaling over IP a step closer to an all-IP network", Masters thesis, Royal Institute of Technology (KTH), Dept. of Communication Systems, June 2005

ftp://ftp.it.kth.se/Reports/DEGREE-PROJECT-REPORTS/050619-Mia-Immonen-with-cover.pdf

Maguire
maguire@it.kth.se

References
2006.03.13

SCTP 342 of 342
Internetworking/Internetteknik

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 7: Dynamic Routing

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapter 14

KTH Information and Communication Technology

Outline

- Dynamic Routing Protocols

Maguire
maguire@it.kth.se

Outline
2006.03.13

Dynamic Routing 344 of 413
Internetworking/Internetteknik
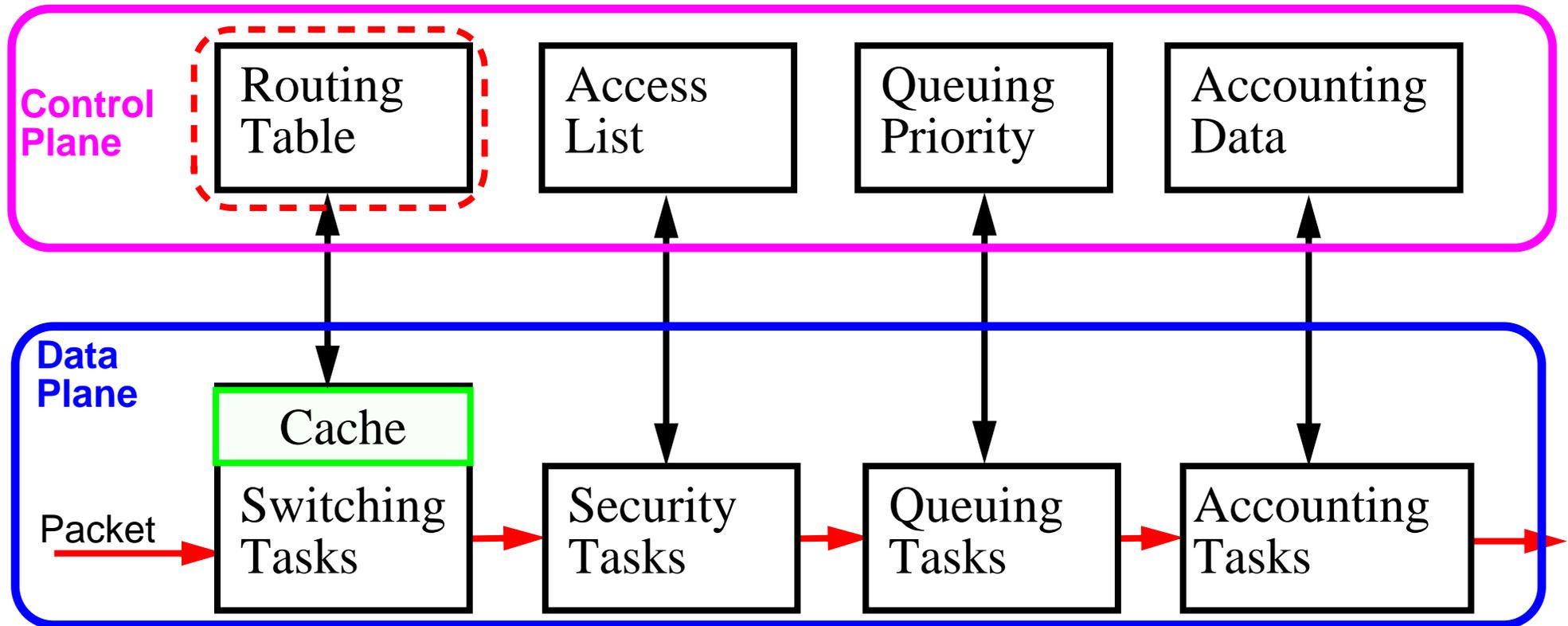
Routing

Figure 56: Basic steps in Routing

The routing table tells us which output port to use based on the destination (and possibly the source) IP address. The data plane has to run at packet rates (i.e., in real-time). However, a router also performs a lot of other processing

Maguire
maguire@it.kth.se

Routing
2006.03.13

Dynamic Routing 345 of 413
Internetworking/Internetteknik

Routing Principles

- Routing Mechanism: Use the *most specific* route
  - IP provides the mechanism to route packets
- Routing Policy: What routes should be put in the routing table? <<< today's topic!
  - Use a routing daemon to provide the *routing policy*

For further information see:

Bassam Halabi, *Internet Routing Architectures*, Cisco Press, 1997, ISBN 1-56205-652-2. -- especially useful for IGRP.

Maguire
maguire@it.kth.se

Routing Principles
2006.03.13

Dynamic Routing 346 of 413
Internetworking/Internetteknik

Basic Router Software Architecture
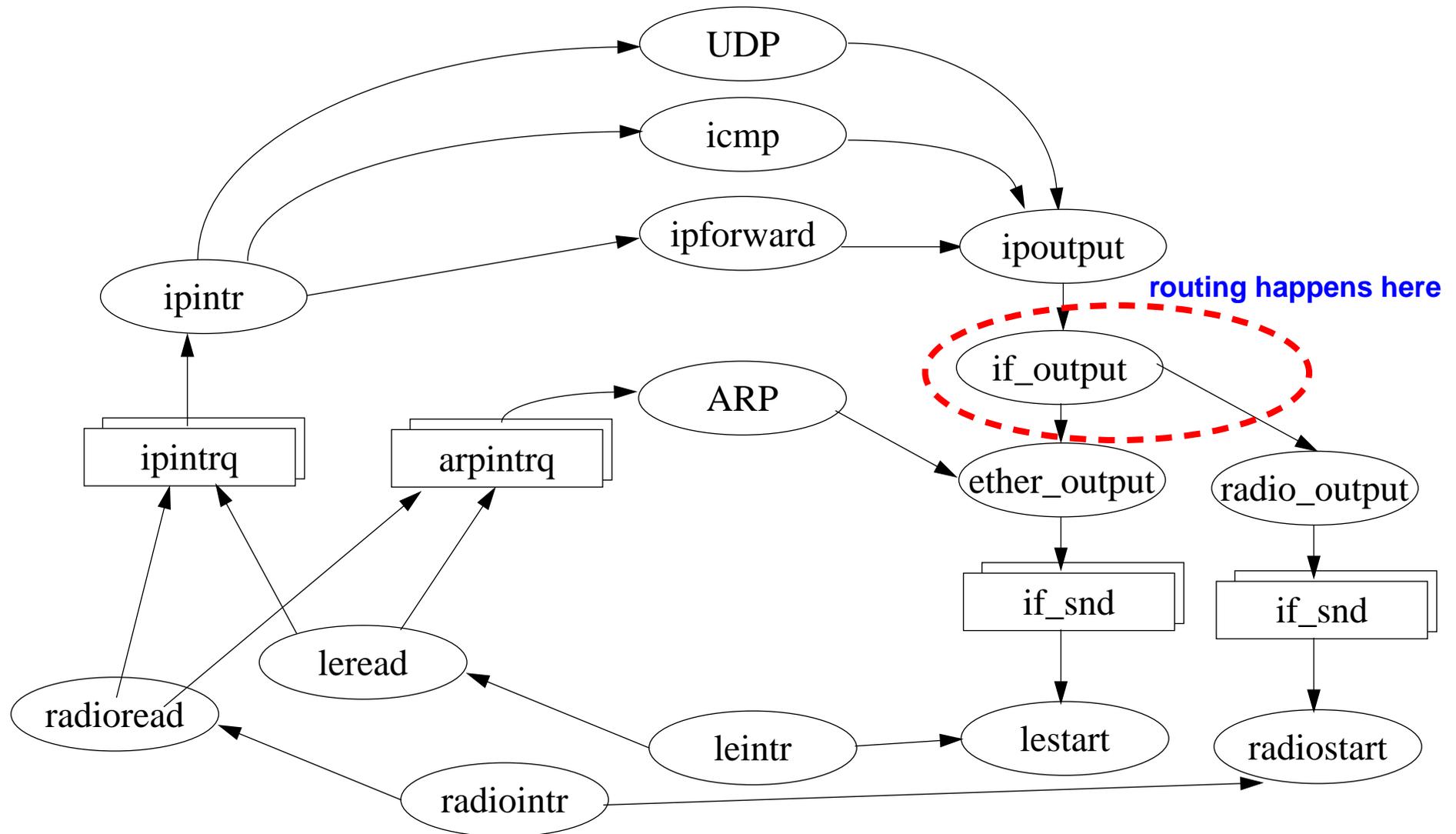
Figure 57: Basic Router Architecture (assuming a radio and an ethernet interface)

Maguire
maguire@it.kth.se

Basic Router Software Architecture
2006.03.13

Dynamic Routing 347 of 413
Internetworking/Internetteknik

Processing

Routing Policy

Rouing daemon	route command	netstat command

UDP   TCP

Routing Mechanism

routing
table
update
from
adjacent
routers

ICMP

Yes

No   our packet
or a broadcast

forward datagram
(if forwarding is enabled)

ICMP
redirects

Routing
Table

IP Output:

calculate next hop router
(if necessary)

source

routing

process IP options

IP input queue

IP Layer

network interfaces

Maguire
maguire@it.kth.se

Processing
2006.03.13

Dynamic Routing 348 of 413
Internetworking/Internetteknik

Routing packets in the Internet

Router needs to know where to route packets, to do this they need routing information. Such information can be provided by **manually entered routes** or ICMP Redirect or learning of routes via **a routing protocol**.

Dynamic routing protocols are based on routers talking to each other.

- Intradomain - <u>within</u> an AS (aka Interior Gateway protocols)
- Interdomain - <u>between</u> ASs (aka Exterior Gateway protocols)

The most popular dynamic routing protocols are:

- RIP-1 - Routing Information Protocol (version 1)
- RIP-2 - Routing Information Protocol (version 2)
- OSPF - Open Shortest Path First
- BGP - Border Gateway Protocol

Maguire
maguire@it.kth.se

Routing packets in the Internet
2006.03.13

Dynamic Routing 349 of 413
Internetworking/Internetteknik

Autonomous systems (ASs) - RFC1930 [38]

Each of which is generally administered by a **single entity**.

Each autonomous system selects the routing protocol to be used **within** the AS.

Network	AS number
Swedish University Network (SUNET)	AS1653 and AS2859
SUNET-KI	AS2837
Stockholm University - SU	AS2838
SUNET-KTH	AS2839
KTHNOC KTHNOC-SE	AS3224

For statistics about the number of AS, etc.: *http://www.cidr-report.org/*

For a list of AS number to name mappings: *http://www.cidr-report.org/autnums.html*

To find out who is responsible for a given autonomous system, use a query of the form: *http://www.ripe.net/perl/whois?AS2839*

Maguire
maguire@it.kth.se
Autonomous systems (ASs) - RFC1930 [38]
2006.03.13
Dynamic Routing 350 of 413
Internetworking/Internetteknik

Routing Metrics

A measure of which route is better than another:

- Number of hops
- Bandwidth
- Delay
- Cost
- Load
- …

It is possible that the metric uses some **weighted combination** of the above.

Maguire
maguire@it.kth.se

Routing Metrics
2006.03.13

Dynamic Routing 351 of 413
Internetworking/Internetteknik

Routing Algorithms

- Static vs. Dynamic
- Single path vs. Multi-path
- Flat vs. Hierarchical
- Host-intelligent vs. Router-intelligent
- Intradomain (interior) vs. Interdomain (exterior)
- Link state vs. Distance vector

Issues:

- Initialization (how to get started)
- Sharing
- Updating
- When to share & Who to share with

Maguire
maguire@it.kth.se

Routing Algorithms
2006.03.13

Dynamic Routing 352 of 413
Internetworking/Internetteknik

Intradomain routing protocols

also called "Interior Gateway protocols"

Examples:

- HELLO - an old IGP protocol
- RIP - widely used
- OSPF - increasingly used

Routers that speak any dynamic routing protocol **must** speak RIP and OSPF.

Maguire
maguire@it.kth.se

Intradomain routing protocols
2006.03.13

Dynamic Routing 353 of 413
Internetworking/Internetteknik

Routing Information Protocol (RIP) version 1

RFC 1058 [39] {Note it was written years after the protocol was in wide use!}

RIP is a distance-vector protocol - RIP messages contain a vector of **hop counts**. RIP messages are carried via UDP datagrams which are **broadcast**.

0 7	8 Version = 1 15	16 Reserved 31
Command	Version = 1	Reserved
Family		All 0s
Network Address		
All 0s		
All 0s		
Distance		

Figure 58: RIP message format (see Forouzan figure 14.9 pg. 394)

- a command: request or reply

- a version number (in this case 1)

- up to 25 instances (entries) containing:
  - address family (2 = IP addresses)
  - Network Address (allocated 14 bytes, for an IP address we only need 4 bytes - and they are aligned to a 4 byte boundary - hence the leading and trailing zeros)
  - metric [hop count]

Maguire
maguire@it.kth.se

Routing Information Protocol (RIP) version 1
2006.03.13

Dynamic Routing 354 of 413
Internetworking/Internetteknik

RIP v1 operation

As carried out by UNIX daemon "routed" using UDP port 520

Initialization:

```
for all interface which are up
{
  send a request packet out each interface asking for the other
  router's complete routing table
          [command=1, address family=0 {== unspecified}¹, metric=16
}
```

Request received:

```
if whole table requested, then send it all 25 at a time
else if a specific set of routes
      then fill in the metric
      else set metric to 16
          [16 == "infinity" == we don't know a route to this address]
```

Response received:

```
if valid (i.e., not 16),then update/add/delete/modify routing table
```

---

1. Page 24 of RFC 1058 says "If there is exactly one entry in the request, with an address family identifier of 0 (meaning unspecified), and a metric of infinity (i.e., 16 for current implementations), this is a request to send the entire routing table."[39] - this is different than implied in Forouzan figure 14.10 pg. 395

Maguire
maguire@it.kth.se
RIP v1 operation
2006.03.13
Dynamic Routing 355 of 413
Internetworking/Internetteknik

When are routes sent?

Solicited response: Send a response when a request is received

Unsolicited response:

- If a metric for a route changes, then (trigger) send update, else send all or part of the table every 30 seconds.
- If a route has not been updated for 180 seconds (3 minutes = 6 update cycles), then set metric to 16 and then **after** 60 seconds (1 minute) delete route.

Metrics are in units of hops, thus this protocol leads to selection between routes based on the minimum number of hops.

Summary of RIPv1 Timers:

- Periodic timer - regular updates random value [25..35] mean 30 s.
- Expiration timer - routes not updated within (180 s) expire
- Garbage collection timer - 120 s after expiring entries are GC'd

Maguire
maguire@it.kth.se
RIP v1 operation
2006.03.13
Dynamic Routing 356 of 413
Internetworking/Internetteknik

Problems with RIP v1

- RIPv1 does **not** know about subnets (or assumes all interfaces on the network have the same netmask)

- after a router or link failure RIP takes **minutes** to stabilize (since each neighbor only speaks ~every 30 seconds; so the time for the information to propagate several hops is minutes) $\Rightarrow$ while it is unstable it is possible to have routing loops, etc.

- Hops count may **not** be the best indication for which is the best route

- Since the maximum useful metric value is 15, the network diameter must be less than or equal to 15.

- RIP will accept updates from anyone - so one misconfigured device can disrupt the entire network.

- RIP uses more bandwidth than other protocols, since it sends the whole routing table in updates.

Maguire
maguire@it.kth.se

Problems with RIP v1
2006.03.13

Dynamic Routing 357 of 413
Internetworking/Internetteknik

Count to Infinity
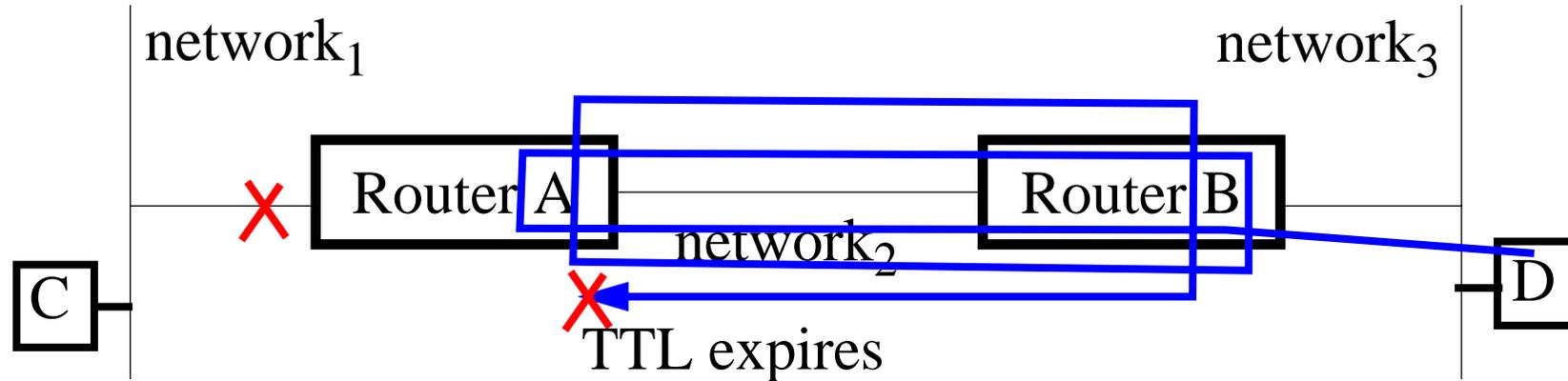
- Router A advertises it knows about routes to networks 1 and 2
- Router B advertises it knows about routes to networks 2 and 3
- After one update cycles A and B know about all 3 routes.

If A's interface to Network$_1$ goes down, then A learns from B - that B knows a route to Network$_1$; so A now thinks it can reach Network$_1$ via B. So if D sends a packet for C, it will simply loop back and forth between routers A and B, until the TTL counts down to 0.

Maguire
maguire@it.kth.se

Count to Infinity
2006.03.13

Dynamic Routing 358 of 413
Internetworking/Internetteknik

Split Horizon

To counter the count to infinity, the split horizon algorithm - never sends information on an interface that it learned from this interface.

RIPv1 implements: Split Horizon with Poison Reverse Update - rather that not advertise routes to the source, we advertise them with a metric of 16 (i.e., unreachable) - hence the source simply ignores them.

Unfortunately split horizon only prevents loops between **adjacent** routers (so if there are three or more routers involved the previous problem re-appears)

Maguire
maguire@it.kth.se

Split Horizon
2006.03.13

Dynamic Routing 359 of 413
Internetworking/Internetteknik

Triggered updates and Hold-Downs

To decrease convergence time - when the topology changes send out an update immediately.

However, if a node can learn about connectivity from more than one source, then if a delete happens before the add, then the existence of the route is asserted; therefore the hold-down rules says:

> When a route is removed, no update of this route is accepted for some period of time - to give everyone a chance to remove the route.

> This period of time is the Hold-down time.

The result is to decrease the rate of convergence; but the combined effect of triggered updates + hold-downs leads to **faster** convergence than not using triggered updates.

Maguire
maguire@it.kth.se

Triggered updates and Hold-Downs
2006.03.13

Dynamic Routing 360 of 413
Internetworking/Internetteknik

RIP extensions (aka RIP-2)

Defined in RFC 1388 [40] and revised in RFC 2453[41]. Version number is 2.

- for each of up to 25 entries we add the fields:
  - Route tag - carries the AS number
  - Subnetmask - subnetmask to be used with this address (to support classless addressing)
  - Next-hop IP address, either the IP address of where packets to this destination should be sent or zero [which means send them to the system which sent the RIP message]

- One entry can be replaced by Authentication data

RIP-2 supports multicast to address 224.0.0.9, to reduce load on hosts **not** interested in RIP-2 messages

0 7	8 15	16 23	24 31
Command	Version = 2	Reserved	
0xFFFF		Authentication type	
Authentication data (16 bytes) if Authentication type = 2, this is a clear text password to be used to authenticate this message			
Family		Route tag	
Network Address			
Subnet mask			
Next-hop address			
Distance			

Figure 59: RIPv2 message format (see Forouzan figures 14.13 pg. 397 and 14.14 pg. 398)

Maguire
maguire@it.kth.se

RIP extensions (aka RIP-2)
2006.03.13

Dynamic Routing 361 of 413
Internetworking/Internetteknik

Why would anyone use RIP?

After all these problems you might ask this question.

Answer

- Because RIP is generally the only routing protocol which **all** UNIX machines understand!

- Relatively easy to configure

- It it widely available, since it **must** exist if the device is capable of routing!

Maguire
maguire@it.kth.se

Why would anyone use RIP?
2006.03.13

Dynamic Routing 362 of 413
Internetworking/Internetteknik

Interior Gateway Routing Protocol (IGRP)

Cisco's IGRP [42] - a proprietary protocol with the following goals:

- stable, optimal routing for large networks - with no routing loops
- fast response to changes in net topology
- low overhead in both bandwidth and processor utilization
- ability to split traffic across several parallel routes if they are (or nearly are) equal.

It is a distance-vector protocol based on many of the ideas from RIP.

Maguire
maguire@it.kth.se

Interior Gateway Routing Protocol (IGRP)
2006.03.13

Dynamic Routing 363 of 413
Internetworking/Internetteknik

IGRP Metrics

- a vector of metrics each with a 24 bit value

- composite metric is $\left[\left(\dfrac{K_1}{B}\right) + \left(\dfrac{K_2}{D}\right)\right] \cdot R$, where K_1 and K_2 are constants, B the unloaded path bandwidth, D a topological delay, and R is reliability

- also we pass the hop count and Maximum Transmission Unit values

K_1 is the weight assigned to bandwidth (by default 10,000,000)

K_2 is the weight assigned to delay (by default 100,000)

If up to 4 paths are with in a defined variance of each other, Cisco's IOS (Internetwork Operating System) will split the traffic across them in inverse proportion to their metric.

Maguire
maguire@it.kth.se

IGRP Metrics
2006.03.13

Dynamic Routing 364 of 413
Internetworking/Internetteknik

IGRP Route Poisoning

IGRP poisons routes which increase by a factor of 10% or more after an update [they are thought to be: "too good to be true"].

While this rule may temporarily delete valid routes (which will get reinstated after the next regular update) - it allows use of a zero hold-down time, which leads to faster convergence.

Maguire

maguire@it.kth.se

IGRP Route Poisoning

2006.03.13

Dynamic Routing 365 of 413

Internetworking/Internetteknik

IGRP Default Gateway

Rather than using the fake network 0.0.0.0 to indicate the default network, IGRP allows a real network to be flagged as the default network.

Periodically, IGRP scans the routes offering a path to this flagged network and selects the path with the lowest metric.

Note: Default gateways help to keep the size of local routing tables smaller.

Maguire
maguire@it.kth.se

IGRP Default Gateway
2006.03.13

Dynamic Routing 366 of 413
Internetworking/Internetteknik

Enhanced IGRP (EGRP) [43]

Uses the distance-vector technology of IGRP, but changes the way routes are advertised and the calculation of entries for the routing table.

EGRP uses:

- a neighbor discover/recovery process of hello packets to learn its neighbors
- a Reliable Transport Protocol to ensure guaranteed, ordered delivery of routing updates
- a Diffusing Update Algorithm (DUAL) - which selects both the best route for insertion into the table **and** a feasible successor (for if the primary route fails)
- Variable length subnet masks (VLSM)

EGRP is a Cisco proprietary technology.

Maguire
maguire@it.kth.se

Enhanced IGRP (EGRP) [43]
2006.03.13

Dynamic Routing 367 of 413
Internetworking/Internetteknik

Open Shortest Path First (OSPF)

OSPF defined in RFC2328 (see also *http://rtg.ietf.org/wg/ospf/*)

OSPF is a link-state protocol. OSPF messages (Link State Advertisements (LSAs)) tell the **status of links** of each of its neighbors and propagates this info to its neighbors. Each router uses this link-state information to build a **complete** routing table. Uses IP directly (protocol field = OSPF (89)) $\Rightarrow$ does **not** use UDP or TCP.

Advantages

- link-state protocols converge faster than distance-vector protocols
- can calculate a route per IP service type (i.e., TOS)
- each interface can have a per TOS cost
- if there are several equally good routes $\Rightarrow$ can do load balancing
- supports variable length subnet masks
- enable point to point links to be unnumbered (i.e., don't need an IP address)
- uses clear text passwords
- uses multicasting

Maguire
maguire@it.kth.se

Open Shortest Path First (OSPF)
2006.03.13

Dynamic Routing 368 of 413
Internetworking/Internetteknik

OSPF uses the Shortest Path First algorithm (also know as Dijkstra's algorithm).

OSPF networks generally divided into areas such that cross-area communication is minimal.

Some routers with multiple interfaces become border area routers (with one interface in one area and another interface in another area).

The only way to get from one area to another area is via the backbone - which is area 0. Note: The backbone need **not** be continuous.

- Note that Forouzan refers to "transient" links -- I think that this should be "transit" links. (Since transient implies that the link would be short lived!)

Link state advertisements are sent to all routers in a given area (via flooding), rather than just neighbors (as in the distance-vector approach) - thus periodic updates are infrequent (every 1 to 2 hours).

A key feature of OSPF is route aggregation - which minimizes the size of routing tables and the size of the topological database; in addition, it keeps protocol traffic to a minimum.

Maguire
maguire@it.kth.se
Open Shortest Path First (OSPF)
2006.03.13
Dynamic Routing 369 of 413
Internetworking/Internetteknik

OSPF building blocks

1. Hello protocol

- Check for & with neighbors and learn designated routers

2. Synchronization of Databases

- Exchange of Link State Database between neighbors
  - Get LSA **headers**
  - Request the transfer of necessary LSAs

3. Flooding protocol

- When links change or when your knowledge is old
  - Send Link State updates to neighbors and flood recursively
  - If not seen before, propagate updates to all adjacent routers, except the router you received it from

Maguire
maguire@it.kth.se

OSPF building blocks
2006.03.13

Dynamic Routing 370 of 413
Internetworking/Internetteknik

OSPF Packets

Common header

0 7	8 15	16 23 24 31
Version	Type	Message length
Source Router IP Address		
Area Identification		
Checksum		Authentication type
Authentication (64 bits)[a]		

Figure 60: OSPF Common Header (see Forouzan figure 14.26 pg. 408)

a. Note that the Authentication field is 64 bits, Forouzan figure 14.26 pg. 408 incorrectly shows it as 32 bits.

The 5 types of OSPF packets:

Type	Description
1	Hello
2	Database Description
3	Link State Request
4	Link State Update
5	Link State Acknowledgment

Maguire
maguire@it.kth.se

OSPF Packets
2006.03.13

Dynamic Routing 371 of 413
Internetworking/Internetteknik

Hello packet

0	7	8	15	16	23	24	31
Version		Type = 1		Message length			
Source Router IP Address							
Area Identification							
Checksum				Authentication type			
Authentication (64 bits)							
Network Mask							
Hello interval (seconds)				All zeros	E	T	Priority
Dead interval (seconds)							
Designated router IP address							
Backup designated router IP address							
Neighbor IP address							
…							
Neighbor IP address							

Figure 61: OSPF Link state update packet (see Forouzan figure 14.44 pg. 419)

- E = 1 indicates a stub area
- T = 1 indicates router supports multiple metrics
- priority = 0 indicates that this router should not be considered as a designated or backup designated router
- Dead interval is the time before a silent neighbor is assumed to be dead
- list of neighboring routers (of the router which sent the hello packet)

Maguire
maguire@it.kth.se

Hello packet
2006.03.13

Dynamic Routing 372 of 413
Internetworking/Internetteknik

Database Description packet

0	7	8	15	16	23	24	31
Version		Type = 2		Message length			
Source Router IP Address							
Area Identification							
Checksum				Authentication type			
Authentication (64 bits)							

Interface MTU		All zeros	E	B	All zeros	I	M	M S

Database Description sequence number
LSA header (20 bytes)
…
LSA header

Figure 62: OSPF Database Description packet (see Forouzan figure 14.45 pg. 420)

- ## Rather than send the entire database - send an outline of it
  - E = 1 indicates the advertising router is an autonomous boundary router (i.e., E ≡ external)
  - B = 1 indicates the advertising router is an autonomous border router
  - I = 1 initialization flag
  - M = 1 ≡ More flag
  - M/S flag: 0=slave, 1=Master
  - Database Description sequence number
  - LSA header(s) - gives information about the link - but **without** details; if details are desired they can be requested

Maguire
maguire@it.kth.se

Database Description packet
2006.03.13

Dynamic Routing 373 of 413
Internetworking/Internetteknik

Link State Announcement (LSA) header

Each different LSA has the same general header:

0	7 8	15 16	23 24	31

Link state age (seconds)	Reserved	E	T	Link State type
Link state ID				
Advertising router				
Link state sequence number				
Link state checksum		Length		

Figure 63: OSPF Link State Advertisement general header format (see Forouzan figure 14.28 pg. 410)

- E = 1 indicates a stub area
- T = 1 indicates router supports multiple metrics

Link state type	Link state ID	Description
1	IP address of the router	Router-LSAs
2	IP address of the designated router	Network-LSAs
3	IP address of the network	Summary-LSAs (IP network)
4	IP address of the border router	Summary to AS border router (BR)
5	IP address of the external network	AS-external-LSAs

- Advertising router - IP address of the router advertising this message
- Link state checksum - a Fletcher's checksum of all but age field

Maguire
maguire@it.kth.se

Link State Announcement (LSA) header
2006.03.13

Dynamic Routing 374 of 413
Internetworking/Internetteknik

- Length of the whole packet in bytes

Maguire
maguire@it.kth.se

Link State Announcement (LSA) header
2006.03.13

Dynamic Routing 375 of 413
Internetworking/Internetteknik

Link state update packet

0 7	8 15	16 23 24 31
Version	Type = 4	Message length
Source Router IP Address		
Area Identification		
Checksum		Authentication type
Authentication (64 bits)		
Number of link state advertisements		
Link state advertisement (LSA)		
…		
Link state advertisement (LSA)		

Figure 64: OSPF Link state update packet (see Forouzan figure 14.27 pg. 409)

Maguire
maguire@it.kth.se

Link state update packet
2006.03.13

Dynamic Routing 376 of 413
Internetworking/Internetteknik

Link state request packet

0 7	8 15	16 23 24 31	
Version	Type = 3	Message length	
Source Router IP Address			
Area Identification			
Checksum		Authentication type	
Authentication (64 bits)			
Link state type			repeat
Link state ID			
Advertising router			
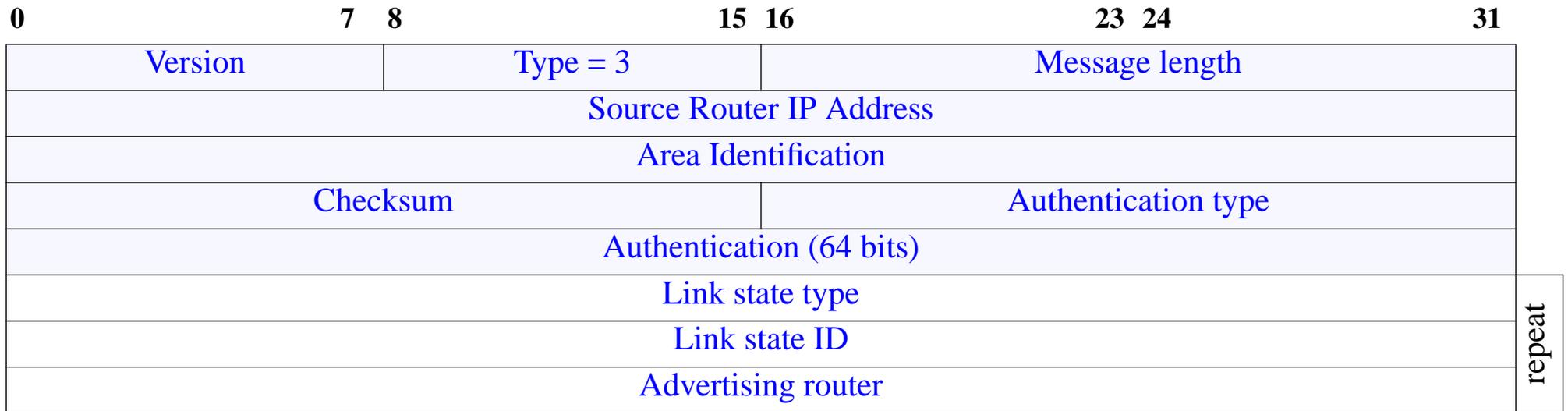
Figure 65: OSPF Link state request packet (see Forouzan figure 14.46 pg. 420)

To ask for information about a specific route (or routes); reply is an update packet.

Maguire
maguire@it.kth.se

Link state request packet
2006.03.13

Dynamic Routing 377 of 413
Internetworking/Internetteknik

Link state acknowledgement packet

0	7	8	15	16	23	24	31
Version		Type = 5		Message length			
Source Router IP Address							
Area Identification							
Checksum				Authentication type			
Authentication (64 bits)							
LSA header							

Figure 66: OSPF Link state acknowledgement packet (see Forouzan figure 14.47 pg. 421)

To acknowledge receipt of an update packet.

Maguire
maguire@it.kth.se

Link state acknowledgement packet
2006.03.13

Dynamic Routing 378 of 413
Internetworking/Internetteknik

Interdomain routing protocols

also called "Exterior Gateway Protocols (EGPs)" - used between ASs

Examples:

- EGP - an old EGP protocol
- BGP - Border Gateway Protocol

Intradomain routing protocols:

- don't scale up to the large numbers of routers involved in interdomain routing (due to the huge computational resources required)
- distance vector routing becomes unstable beyond a few hops

Maguire
maguire@it.kth.se

Interdomain routing protocols
2006.03.13

Dynamic Routing 379 of 413
Internetworking/Internetteknik

Exterior Gateway Protocol (EGP)

an exterior gateway protocol with three components:

- neighbor acquisition
- neighbor reach ability, and
- routing information

EGP was designed to provide more automation in configuring routers.

EGP is similar to the distance-vector protocols, but omits the metrics, since EGP was designed for the internet where typically routers are connected to a backbone (with its own routing domain) via a single router.

- But since there are no metrics, if there is more than one path, then there can be a loop!

Maguire
maguire@it.kth.se

Exterior Gateway Protocol (EGP)
2006.03.13

Dynamic Routing 380 of 413
Internetworking/Internetteknik

BGP - Border Gateway Protocol

An exterior gateway protocol to exchange routing information between routers in different ASs.

BGP version 3 defined in RFC 1267[46], while version 4 defined in RFC1654[47] and RFC 1771[48].

BGP routers exchange routing information with other BGP routers.

For further information see:

John W. Stewart III, *BGP4: Inter-Domain Routing in the Internet*, Addison-Wesley, 1999, ISBN: 0-201-37951-1 [49]

See also [57] and [58].

Maguire
maguire@it.kth.se

BGP - Border Gateway Protocol
2006.03.13

Dynamic Routing 381 of 413
Internetworking/Internetteknik

Local vs. Transit traffic

Local traffic: originates or terminates in an AS

Transit traffic: is all other traffic

==> 3 types of ASs

stub AS	connected to only on other AS, carries only local traffic
multihomed AS	connected to multiple AS, but refuses transit traffic
transit AS	connected to multiple ASs, carries both local and transit traffic

Maguire
maguire@it.kth.se

Local vs. Transit traffic
2006.03.13

Dynamic Routing 382 of 413
Internetworking/Internetteknik
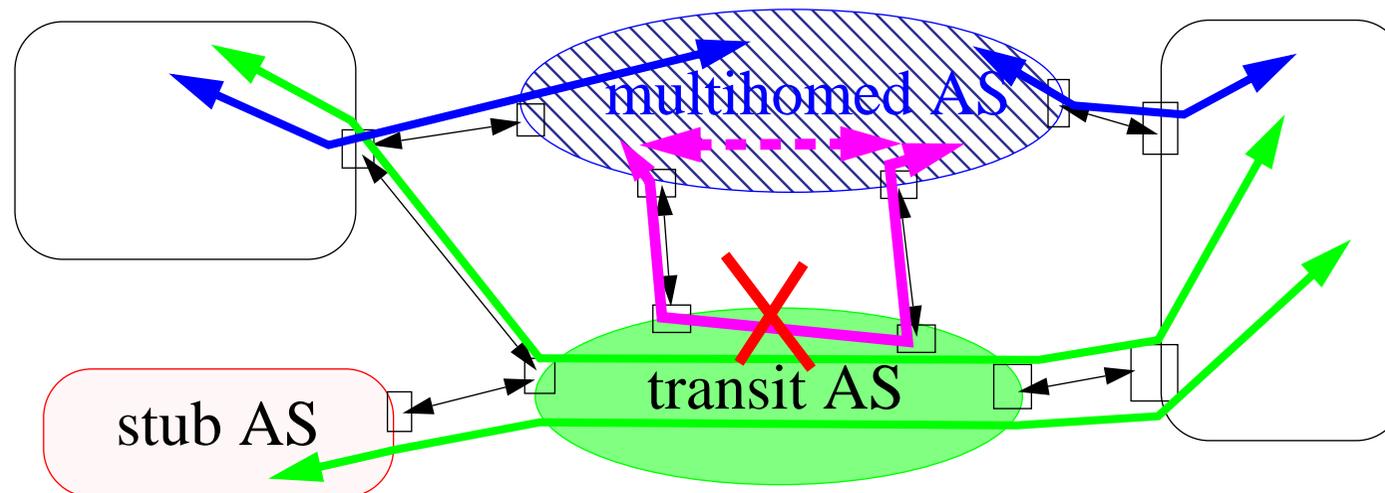
BGP operation

BGP routers exchange information based on traffic which transits the AS, derives a graph of AS connectivity; with loop pruning.

Routing policy decisions can be enforced as to what is allowed to transit whom
$\Rightarrow$ **policy-based routing**

- based on economic/security/political/… considerations.
- BGP does **not** implement the policy decisions, but allows the information on which such decisions can be made to propagate as necessary

Uses TCP (port 179) to create a session between BGP routers:

- initially two systems exchange their entire BGP routing table,
- then they simply send updates as necessary.

BGP is a path-vector protocol - which **enumerates** the route to each destination (i.e., the sequence of AS numbers which a packet would have to pass through from a source to its destination) = a path vector

Maguire
maguire@it.kth.se

BGP operation
2006.03.13

Dynamic Routing 383 of 413
Internetworking/Internetteknik

BGP does **not** transmit metrics.

However, each path is a list of attributes:

- well-known attributes (which every router must understand)
  - well-known mandatory attribute - must appear in the description of a route
  - well-known discretionary attribute - may appear, but must be recognized, in the description of a route
- optional attributes
  - optional transitive attribute - must be passed to the next router
  - optional nontransitive attribute - the receiving router must discarded it if does not recognize it

For examples of the use of an attribute see [55] and [56]

BGP detects failures (either links or hosts) by sending **keepalive** messages to its neighbors. Generally sent every 30 seconds and as they are only 19 bytes each $\Rightarrow$ only ~5 bits/second of bandwidth, but with very long lived TCP connections (semi-permanent connections)

A major feature of BGP version 4 is its ability to do aggregation - to handle CIDR and supernetting. For more information on aggregation see chapter 5 of [50].

Maguire
maguire@it.kth.se
BGP operation
2006.03.13
Dynamic Routing 384 of 413
Internetworking/Internetteknik

Classless Inter-Domain Routing (CIDR)

A standard for both classless addressing and classless interdomain routing scheme (RFCs 1517 [51] .. 1520 [54]).

- Basic concept: to allocate/collapse a block of contiguous IP addresses into a single routing table entry: (network address, count). e.g., 192.5.48.0, 192.5.49.0, 192.5.50.0 = (192.5.48.0, 3)

- Hierarchical Routing Aggregation minimizes routing table entries; enables "route aggregation" in which a single high-level route entry can represent many lower-level routes in the global routing tables.
    - Reduces the growth of routing table.

- Allows the addresses assigned to a single organization to span multiple classed prefixes.

- Envisioned a hierarchical Internet.

CIDR addressing scheme and route aggregation has two major user impacts:

- you have to **justifying** IP Address Assignments
- get address from your ISP, i.e., renting them vs. being assigned them

Maguire
maguire@it.kth.se

Classless Inter-Domain Routing (CIDR)
2006.03.13

Dynamic Routing 385 of 413
Internetworking/Internetteknik

Redistribution of Route Information between protocols

Redistribution: allows a router running more than one routing protocol to distribute information from one protocol to another.

Thus at the border, a router will translate the information obtained from one routing domain and pass it to the other routing domain in the appropriate manner.

- Advertize (aggregated) interior routes to the Internet
- Inject (some) exterior routes into the interior network

Usually the redistributed routes are filtered (as not all the information needs to cross the border).

Maguire
maguire@it.kth.se
Redistribution of Route Information between protocols
2006.03.13
Dynamic Routing 386 of 413
Internetworking/Internetteknik

Resulting BGP Architecture
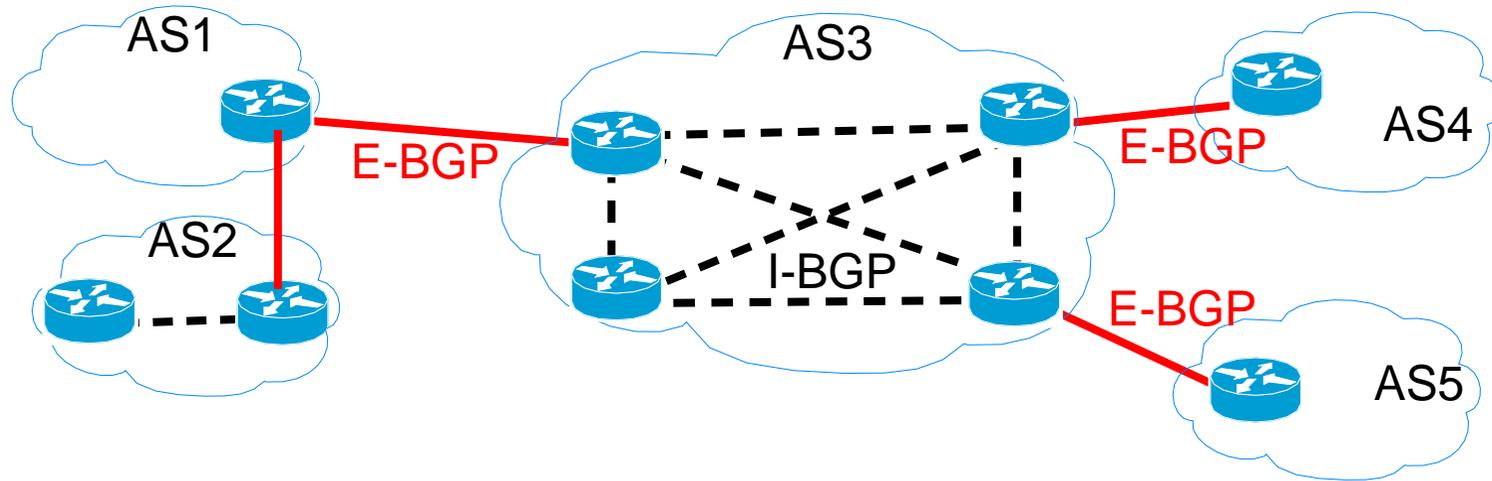
Figure 67: BGP architecture

Two kinds of BGP sessions:

- External (E-BGP) coordinates between border routers between ASs
- Internal BGP (I-BGP) coordinates between BGP peers within an AS
  - Note it must be a full mesh, but this does not scale ⇒ organization into subASs, … .

Maguire
maguire@it.kth.se

Resulting BGP Architecture
2006.03.13

Dynamic Routing 387 of 413
Internetworking/Internetteknik

BGP Messages

- Open
- Update
- Keepalive
- Notification

Maguire

maguire@it.kth.se

BGP Messages

2006.03.13

Dynamic Routing 388 of 413

Internetworking/Internetteknik

BGP Open Message

Marker (16 bytes)			
Length		Type = 1	Version
My autonomous system		Hold time	
BGP identifier			
Option length			
Option (variable length)			
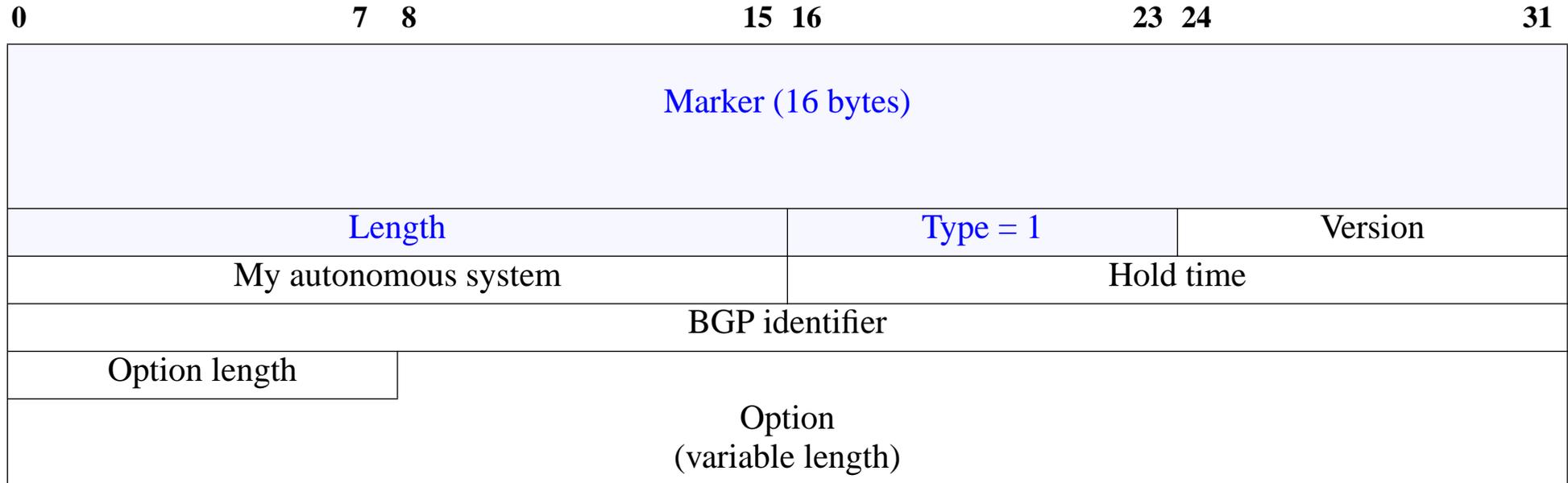
Figure 68: BGP Open messages (see Forouzan figures 14.53 pg. 427 and 14.52 pg. 426)

Maker (for authentication), Length, and Type are common to all BGP messages.

- Version = 4
- My autonomous system - the AS number
- Hold time - maximum time to wait for a keepalive or update, otherwise the other party is considered to be dead
- BGP identifier - identifies the router sending this message (typically its IP address)
- Option length - zero if none
- Option - options in the form (length of parameter, parameter value)

Maguire
maguire@it.kth.se

BGP Open Message
2006.03.13

Dynamic Routing 389 of 413
Internetworking/Internetteknik

BGP Update Message

0		7	8		15	16		23	24		31

Marker (16 bytes)

Length	Type = 2	Unfeasible routes length
Unfeasible routes length		

Withdraw routes
(variable length)

Path attributes length

Path attributes
(variable length)

Network layer reachability information
(variable length)
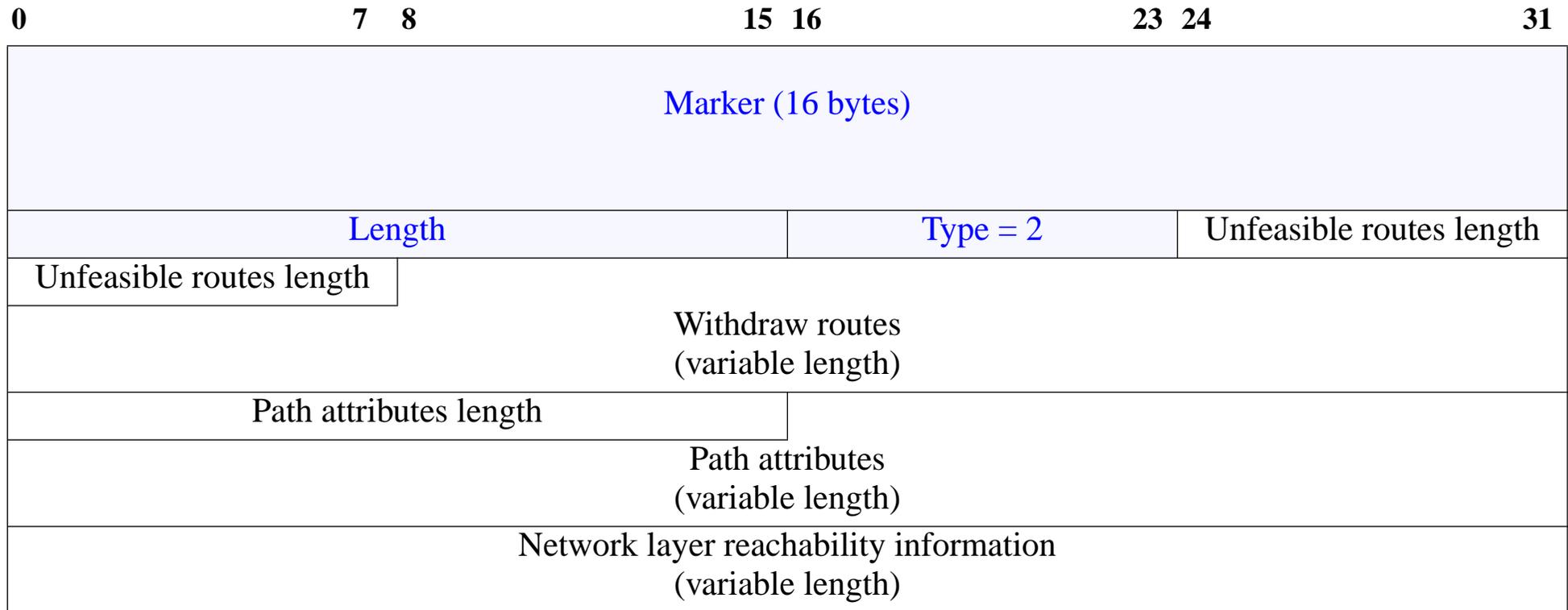
Figure 69: BGP Update message (see Forouzan figures 14.54 pg. 428 and 14.52 pg. 426)

- Unfeasible routes length (2 bytes) - length of next field
- Withdraw routes - list of all routes that must be deleted
- Path attributes length(2 bytes) - length of next field
- Path attributes - specifies the attributes of the path being announced
- Network layer reachability information - prefix length and IP address (to support CIDR)

Maguire
maguire@it.kth.se

BGP Update Message
2006.03.13

Dynamic Routing 390 of 413
Internetworking/Internetteknik

BGP Keepalive Message

Marker (16 bytes)		
Length	Type = 3	

Figure 70: BGP Keepalive message (see Forouzan figures 14.55 pg. 429 and 14.52 pg. 426)

- Sent to reassure your peer that you are still alive

Maguire
maguire@it.kth.se

BGP Keepalive Message
2006.03.13

Dynamic Routing 391 of 413
Internetworking/Internetteknik

BGP Notification Message

Marker (16 bytes)			
Length		Type = 4	Error code
Error subcode			
Error Data (variable length)			

Bits: 0 — 7 — 8 — 15 — 16 — 23 — 24 — 31
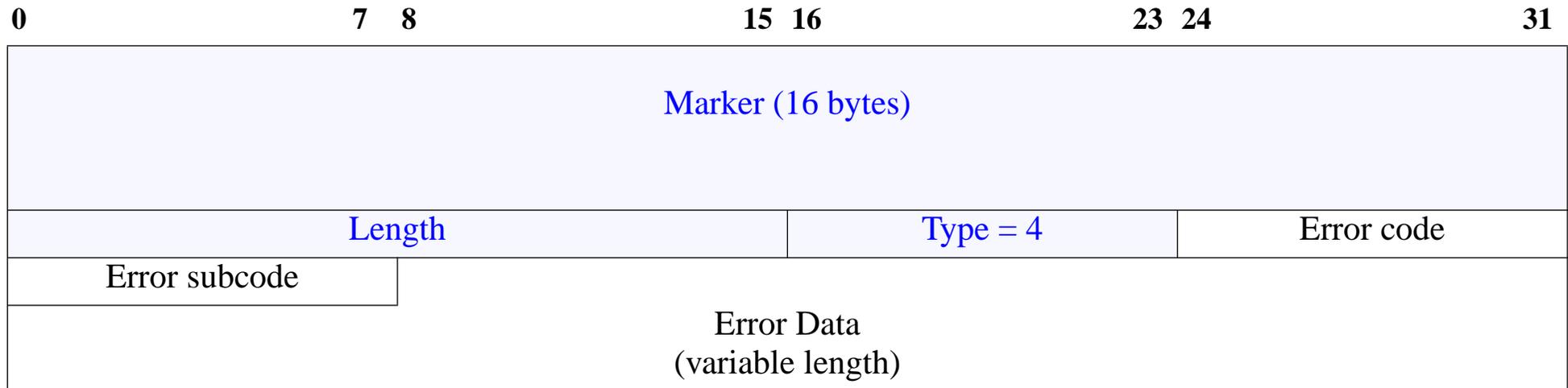
Figure 71: BGP Notification message (see Forouzan figures 14.56 pg. 429 and 14.52 pg. 426)

- Error code (1 bytes) - category of error
  - 1 = Message header error
  - 2 = Open Message error
  - 3 = Update Message error
  - 4 = Hold time expired
  - 5 = Finite state machine error
  - 6 = Cease
- Error subcode (1 bytes) - the particular error in this category
- Error Data - information about this error

Maguire
maguire@it.kth.se

BGP Notification Message
2006.03.13

Dynamic Routing 392 of 413
Internetworking/Internetteknik

Interconnections of networks

Since different networks have different users, policies, cost structure, etc.

But, the value of a network is proportional to the (number users)^2 [Metcalf's Law]

Therefore, network operators **want** to connect their networks to other networks.

$\Rightarrow$ Internet eXchange Points (IXs or IXPs)

List of public internet exchange points: *http://www.ep.net/ep-main.html*

For a discussion of why IXPs are important see [59]

- No internet exchange points $\Rightarrow$ no internetworking!
- Cost advantages in peering
- QoS advantages

Federal Internet eXchange (FIX)

A top-level routing domain - i.e., it does not use default routes.

Each was built around an FDDI ring which interconnected routers from each of the operators.

Each of these routers was in turn connected to the rest of the operator's network via a high speed link (often at speeds up to 45Mbps).

Note that it need not be a physical ring, but was often an FDDI switch (such as the DEC Gigaswitch/FDDI).

Maguire
maguire@it.kth.se

Federal Internet eXchange (FIX)
2006.03.13

Dynamic Routing 394 of 413
Internetworking/Internetteknik

Commercial Internet eXchange (CIX)

A nonprofit trade association of Public Data Internetwork Service Providers.

- a neutral forum - for forming consensus on legislation and policies
- fundamental agree for all CIX members to interconnect with on another
- no restriction on traffic between member networks
- no "settlements" or traffic charges

Maguire
maguire@it.kth.se

Commercial Internet eXchange (CIX)
2006.03.13

Dynamic Routing 395 of 413
Internetworking/Internetteknik

Global Internet eXchange (GIX)

Global Internet eXchange (GIX), Guy Almes, Peter Ford, Peter Lothberg.

proposed in June 1992 - Stockholm D-GIX became Netnod-IX

Maguire
maguire@it.kth.se

Global Internet eXchange (GIX)
2006.03.13

Dynamic Routing 396 of 413
Internetworking/Internetteknik

Some of Sweden's Internet exchange points

- Luleå Internet Exchange ⇒ Polarix *http://www.polarix.se/*
- NorrNod *http://www.norrnod.se/*
- NETNOD Internet eXchange *http://www.netnod.se/*
- RIX -GH Gävleborg Regional Internet Exchange *http://www.rix-gh.se/*
- SOL-IX - Stockholm *http://www.sol-ix.net/*

Other useful contacts:

SNUS (Swedish Network Users Society)

> "... its goal, from the users perspective, to force the evolution and development of the networks and interconnections between networks, to arrange seminars, to exchange information between the members, and to write agreements with companies."

- SOF (Swedish Operators Forum)
- North American Network Operators' Group (NANOG) *http://www.nanog.org/*
- …

Maguire
maguire@it.kth.se

Some of Sweden's Internet exchange points
2006.03.13

Dynamic Routing 397 of 413
Internetworking/Internetteknik

Network Access Points (NAPs)

At the NAP a high-speed network or switch is used to interconnect a number of routers for the purpose of exchanging traffic.

Started with several NSF sponsored NAPs:

- Sprint NAP - in Pennsauken NJ
- PacBell NAP - in San Francisco
- Ameritech Advanced Data Services (AADS) {now SBC} NAP - in Chicago (ATM exchange point) *http://www.aads.net/main.html*
- MAE-East (part of MCI) - in Washington DC *http://www.mae.net/*
- MAE-West (part of MCI) - in San Jose CA

In addition to handling IP packets, NAPs were required to support InterDomain Routing Protocol (IDRP) {the ISO OSI Exterior Gateway Protocol} and route CLNP (ConnectionLess Networking Protocol) packets.

Maguire
maguire@it.kth.se

Network Access Points (NAPs)
2006.03.13

Dynamic Routing 398 of 413
Internetworking/Internetteknik

NAPs today

Using GigE, switch fabrics, resilient packet ring (Spatial Reuse Protocol (SRP)) technology, e.g., Cisco's Dynamic Packet Transport (DPT), … with dedicated fiber connections to/from members.

NAP managers are increasingly concerned about security, reliability, and accounting & statistics.

Various NAP have different policies, methods of dividing costs, fees, co-location of operators equipment at the NAP, etc.

Maguire
maguire@it.kth.se

NAPs today
2006.03.13

Dynamic Routing 399 of 413
Internetworking/Internetteknik

Router Arbiter Project

Router Arbiter (RA) - *http://www.ra.net* (July 1994 through March 1998)

- provide a common database of route information (network topology and policies) [Routing Arbiter Database (RADB) became Routing Assets Database (RADb) *http://www.merit.edu/nrd/services/radb.html*]

- promote stability and manageability of networks

Instead of a full mesh connection between providers, all the providers peer with a central router server. A Router server (RS):

- maintains a database of all information operators need to set their routing policy (written in RIPE 181, see RFC 1786 $\Rightarrow$ Routing Policy Specification Language (RPSL))

  http://www.merit.edu/internet/documents/internet-drafts/draft-blunk-rpslng-08.txt)

- does not forward packets or perform any switching function

- a distributed rover runs at each RS and collects information which the central network management system later queries.

Maguire
maguire@it.kth.se

Router Arbiter Project
2006.03.13

Dynamic Routing 400 of 413
Internetworking/Internetteknik

Internet Routing Registry (IRR)

- a neutral Routing Registry

- set of routing DBs which includes

  - RIPE Routing Registry (European ISPs) - *http://www.ripe.net/db/irr.html*
  - MCI Routing Registry (MCI customers)
  - CA*net Routing Registry (CA*net customers)
  - ANS Routing Registry (ANS customers)
  - JPRR Routing Registry (Japanese ISPs)
  - Routing Arbiter Database (RADB) (all others)

- entries are maintained by each service provider

Internet Performance and Analysis Project (IPMA) IRR Java Interface

http://salamander.merit.edu/ipma/java/IRR.html

Maguire
maguire@it.kth.se

Internet Routing Registry (IRR)
2006.03.13

Dynamic Routing 401 of 413
Internetworking/Internetteknik

Cisco's NetFlow Switching

Routing Table	Access List	Queuing Prioity	Accounting Data

1st Packet →

Switching Tasks	→	Security Tasks	→	Queuing Tasks	→	Accounting Tasks	→

NetFlow Cache

2nd, 3rd, … Packets →

NetFlow Switching Tasks	→	NetFlow Statistics	→	NetFlow Data Export

Figure 72: Cisco's NetFlow Switching

Maguire
maguire@it.kth.se

Cisco's NetFlow Switching
2006.03.13

Dynamic Routing 402 of 413
Internetworking/Internetteknik

Flows

A flow is defined as a "uni-directional stream of packets between a given source network-layer address and port number and a specific destination network-layer address and port number".

Since many application use well known transport-layer port numbers, it is possible to identify **flows per user per application basis**.

There is a well defined Netflow Switching Developer's Interface which allows you to get the statistics concerning the NetFlow cache and the per flow data (the later gives you essentially billing records).

A general introduction to NetFlow Switching is available at

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/12cgcr/switch_c/xcprt3/xcovntfl.htm

Maguire
maguire@it.kth.se

Flows
2006.03.13

Dynamic Routing 403 of 413
Internetworking/Internetteknik

Cisco's **Tag Switching**

Combine routing with the performance of switching, based on the concept of "label swapping", in which units of data (e.g., a packet or a cell) carry a short, fixed length label that tells switching nodes how to process the data.

Tag Edge
Routers

Tag
Switches

Figure 73: Tag Switching

Maguire
maguire@it.kth.se

Cisco's Tag Switching
2006.03.13

Dynamic Routing 404 of 413
Internetworking/Internetteknik
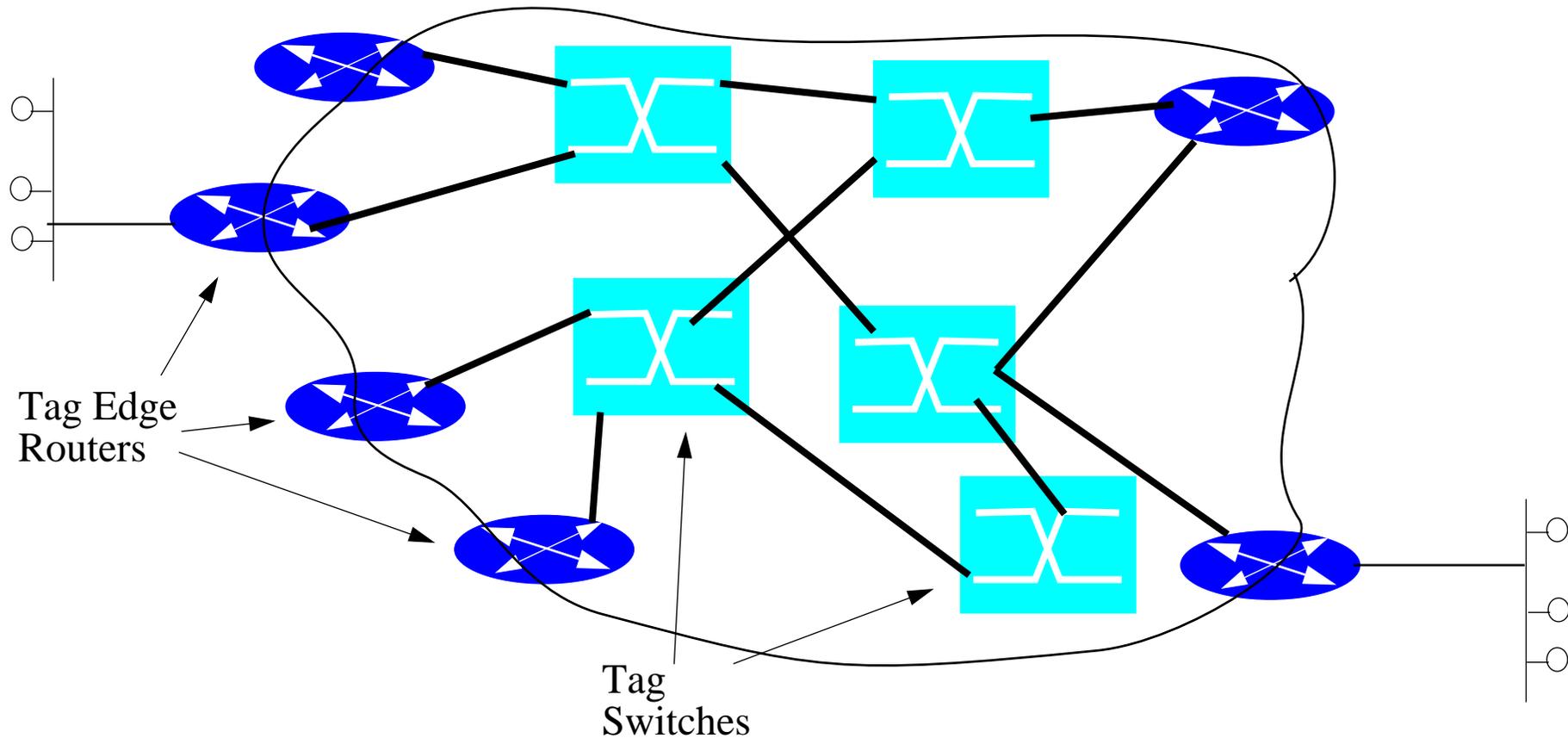
A Tag Edge router labels a packet based on its destination, then the Tag Switches make their switching decision based on this tag, without having to look at the contents of the packet.

The Tag Edge routers and Tag Switch exchange tag data using Tag Distribution Protocol (TDP).

Basics of Tag switching:

1. Tag edge routers and tag switches use standard routing protocols to identify routes through the internetwork.

2. Using the tables generated by the routing protocols the tag edge routers and switches assign and distribute tag information via the tag distribution protocol (TDP). When the Tag routers receive this TDP information they build a forwarding database.

3. When a tag edge router receives a packet it analyzes the network layer header, performs applicable network layer services, selects a route for the packet from its routing tables, applies a tag, and forwards the packet to the next hop tag switch.

4. The tag switch receives the tagged packet and switches the packet based solely on the tag.

5. The packet reaches the tag edge router at the egress point of the network, the tag is stripped off and the packet delivered as usual.

Maguire
maguire@it.kth.se
Cisco's Tag Switching
2006.03.13
Dynamic Routing 405 of 413
Internetworking/Internetteknik

Tag Locations

- in the Layer 2 header (e.g., in the VCI field for ATM cells)
- in the Layer 3 header (e.g., in the flow label field in IPv6) or
- in between the Layer 2 and Layer 3 headers

Maguire
maguire@it.kth.se

Tag Locations
2006.03.13

Dynamic Routing 406 of 413
Internetworking/Internetteknik

Creating tags

Since tag switching decouples the tag distribution mechanisms from the data flows
- the tags can be created:

- when the first traffic is seen to a destination or
- in advance - so that even the first packet can immediately be labelled.

See also: Multiprotocol Label Switching (MPLS) and Generalized Multi-Protocol
Label Switching (GMPLS)

Maguire

maguire@it.kth.se

Creating tags

2006.03.13

Dynamic Routing 407 of 413

Internetworking/Internetteknik

Routers do more than routing
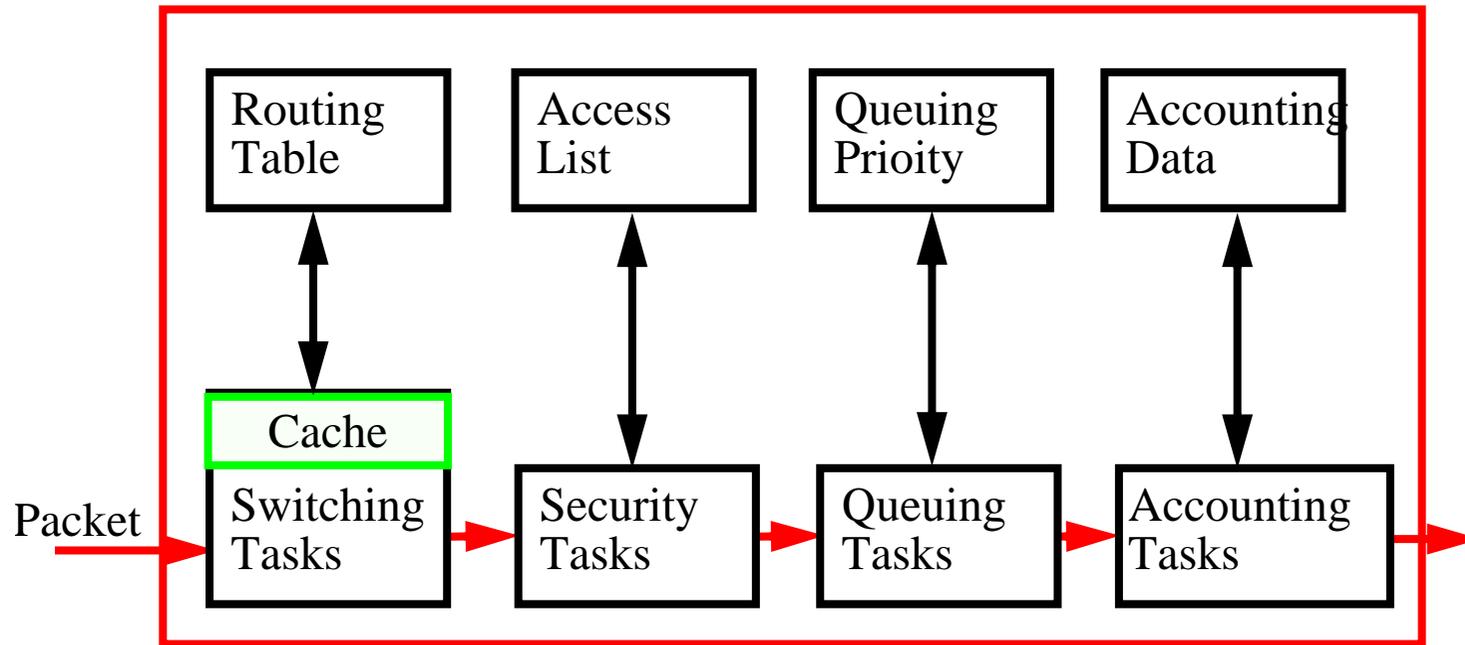
Figure 74: Basic steps in Routing

Earlier we have looked at the routing step, but today much of the excitement is in the details of the other functions. For example, in order to support various QoS features you might want to use more sophisticated queue management: Weighted Round Robin, Fair Queuing, Weighted Fair Queuing, Random Early Detection (RED), Weighted RED, … .

Maguire
maguire@it.kth.se

Routers do more than routing
2006.03.13

Dynamic Routing 408 of 413
Internetworking/Internetteknik

Summary

This lecture we have discussed:

- ## IP routing
  - Dynamic routing protocols
    - RIP, OSPF, BGP, CIDR
  - Cisco's NetFlow Switching and Tag Switching

- ## NAPs and other interconnect points

Maguire
maguire@it.kth.se

Summary
2006.03.13

Dynamic Routing 409 of 413
Internetworking/Internetteknik

References

[38] J. Hawkinson and T. Bates, "Guidelines for creation, selection, and registration of an Autonomous System (AS)", IETF RFC 1930, March 1996 `http://www.ietf.org/rfc/rfc1930.txt`

[39] C. Hedrick, "Routing Information Protocol", IETF RFC 1058, June 1988

`http://www.ietf.org/rfc/rfc1058`

[40] G. Malkin, "RIP Version 2: Carrying Additional Information", IETF RFC 1388, January 1993 `http://www.ietf.org/rfc/rfc1388`

[41] G. Malkin, "RIP Version 2", IETF RFC 2453, November 1998

`http://www.ietf.org/rfc/rfc2453`

[42] Charles L. Hedrick, "An Introduction to IGRP", Cisco, Document ID: 26825, Technology White Paper, August 1991

`http://www.cisco.com/warp/public/103/5.html`

[43] "Enhanced Interior Gateway Routing Protocol", Cisco, Technology White

Maguire
maguire@it.kth.se
References
2006.03.13
Dynamic Routing 410 of 413
Internetworking/Internetteknik

Paper, Document ID: 16406, April 19, 2005

http://www.cisco.com/warp/public/103/eigrp-toc.html

[44] J. Moy, "OSPF Version 2", IETF RFC 2328, April 1998

http://www.ietf.org/rfc/rfc2328.txt

[45] "OSPF Design Guide", Cisco, Technology White Paper, Document ID: 7039,August 26, 2004 *http://www.cisco.com/warp/public/104/1.html*

[46] K. Lougheed and Y. Rekhter, "A Border Gateway Protocol 3 (BGP-3)", IETF RFC 1267, October 1991 *http://www.ietf.org/rfc/rfc1267.tx*t

[47] Y. Rekhter and T. Li (editors), "A Border Gateway Protocol 4 (BGP-4)", IETFRFC 1654,July 1994 *http://www.ietf.org/rfc/rfc1654.txt*

[48] Y. Rekhter and T. Li (editors), "A Border Gateway Protocol 4 (BGP-4)", IETF RFC 1771, March1995 *http://www.ietf.org/rfc/rfc1771.txt*

[49] John W. Stewart III, *BGP4: Inter-Domain Routing in the Internet*, Addison-Wesley, 1999, ISBN: 0-201-37951-1

Maguire
maguire@it.kth.se

References
2006.03.13

Dynamic Routing 411 of 413
Internetworking/Internetteknik

[50] Bassam Halabi, *Internet Routing Architectures*, Cisco Press,
ISBN 1-56205-652-2

[51] R. Hinden (Editor), "Applicability Statement for the Implementation of
Classless Inter-Domain Routing (CIDR)", IETF RFC 1517, September 1993
`http://www.ietf.org/rfc/rfc1517.txt`

[52] Y. Rekhter and T. Li (Editors), "An Architecture for IP Address Allocation
with CIDR", IETF RFC 1518. September 1993 `http://www.ietf.org/rfc/rfc1518.txt`

[53] V. Fuller, T. Li, J. Yu, and K. Varadhan, "Classless Inter-Domain Routing
(CIDR): an Address Assignment and Aggregation Strategy", IETF
RFC 1519, September 1993 `http://www.ietf.org/rfc/rfc1519.txt`

[54] Y. Rekhter and C. Topolcic, "Exchanging Routing Information Across
Provider Boundaries in the CIDR Environment", IETF RFC 1520,
September 1993 `http://www.ietf.org/rfc/rfc1520.txt`

[55] R. Chandra, P. Traina, and T. Li, "BGP Communities Attribute", IETF
RFC 1997, August 1996 `http://www.ietf.org/rfc/rfc1997.txt`

Maguire
maguire@it.kth.se

References
2006.03.13

Dynamic Routing 412 of 413
Internetworking/Internetteknik

[56] E. Chen and T. Bates, "An Application of the BGP Community Attribute in Multi-home Routing", IETF RFC 1998, August 1996

http://www.ietf.org/rfc/rfc1998.txt

[57] Iljitsch van Beijnum, web site http://www.bgpexpert.com/, last modified April 28, 2005 12:00:00 PM

[58] Iljitsch van Beijnum, *BGP*, O'Reilly, 1st Edition September 2002, ISBN 0-596-00254-8

[59] "Internet Exchange Points: Their Importance to Development of the Internet and Strategies for their Deployment: The African Example", Global Internet Policy Iniative (GIPI), 6 June 2002 (revised 3 May 2004),

http://www.internetpolicy.net/practices/ixp.pdf

Maguire
maguire@it.kth.se
References
2006.03.13
Dynamic Routing 413 of 413
Internetworking/Internetteknik

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 8: Multicasting and RSVP

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapters 10, 15

KTH Information and Communication Technology

Outline

- Multicast
- IGMP
- RSVP

Maguire
maguire@it.kth.se

Outline
2006.03.13

Multicasting and RSVP 415 of 493
Internetworking/Internetteknik

Multicast and IGMP

Maguire
maguire@it.kth.se

Multicast and IGMP
2006.03.13

Multicasting and RSVP 416 of 493
Internetworking/Internetteknik

Broadcast and Multicast

Traditionally the Internet was designed for unicast communication (one sender and one receiver) communication.

Increasing use of multimedia (video and audio) on the Internet

- **One-to-many** and **many-to-many** communication is increasing
- In order to support these in a *scalable* fashion we use **multicasting**.
- Replicating UDP packets **where** paths diverge (i.e., split)

MBONE was an experimental multicast network which operated for a number of years. (see for example *http://www-mice.cs.ucl.ac.uk/multimedia/software/* and

http://www.ripe.net/ripe/wg/mbone/home.html)

Multicasting is useful for:

- Delivery to multiple recipients
  - reduces traffic, otherwise each would have to be sent its own copy ("internet radio/TV")
- Solicitation of service (service/server discovery)
  - Not doing a broadcast saves interrupting many clients

Maguire
maguire@it.kth.se

Broadcast and Multicast
2006.03.13

Multicasting and RSVP 417 of 493
Internetworking/Internetteknik

Filtering up the protocol stack
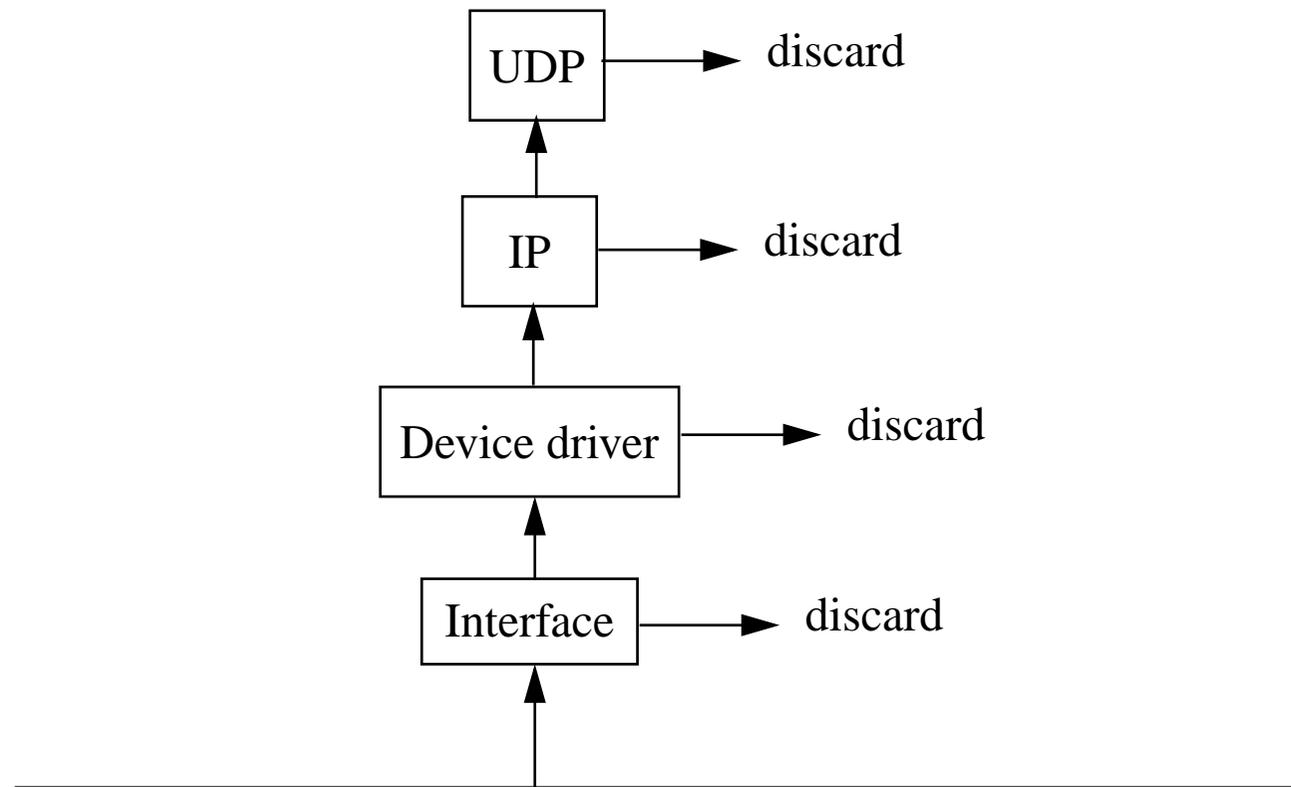
Figure 75: Filtering which takes place as you go up the TCP/IP stack
(see Stevens, Volume 1, figure 12.1, pg. 170)

We would like to filter as soon as possible to avoid load on the machine.

Maguire
maguire@it.kth.se

Filtering up the protocol stack
2006.03.13

Multicasting and RSVP 418 of 493
Internetworking/Internetteknik

Broadcasting

- Limited Broadcast
  - IP address: 255.255.255.255
  - never forwarded by routers
  - What if you are multihomed? (i.e., attached to several networks)
    - Most BSD systems just send on first configured interface
    - routed and rwhod - determine all interfaces on host and send a copy on each (which is capable of broadcasting)

- Net-directed Broadcast
  - IP address: netid.255.255.255 or net.id.255.255 or net.i.d.255 (depending on the class of the network)
  - routers must forward

- Subnet-Directed Broadcast
  - IP address: netid | subnetid | hostID, where hostID = all ones

- All-subnets-directed Broadcast
  - IP address: netid | subnetid | hostID, where hostID = all ones and subnetID = all ones
  - generally regarded as obsolete!

To send a UDP datagram to a broadcast address set SO_BROADCAST

Maguire
maguire@it.kth.se

Broadcasting
2006.03.13

Multicasting and RSVP 419 of 493
Internetworking/Internetteknik

Other approaches to One-to-Many and Many-to-Many communication

Connection oriented approaches have problems:

- large user burden

- have to know other participants

- have to order links in advance

- poor scaling, worst case $O(N^2)$

Maguire
maguire@it.kth.se

Other approaches to One-to-Many and Many-to-Many communication   Multicasting and RSVP 420
2006.03.13
Internetworking/Internetteknik

Alternative centralized model

CU-SeeME uses another model - a Reflector (a centralized model)
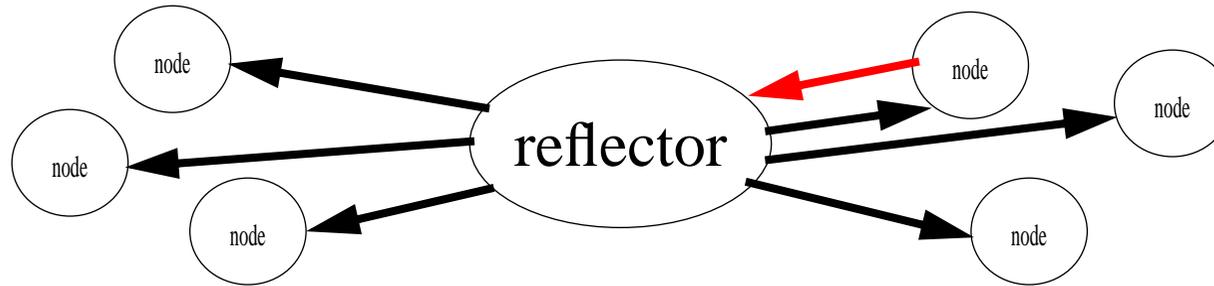
Figure 76: Reflector

- All sites send to one site (the reflector) overcomes the N^2 problems
- The reflector sends copies to all sites

Problems:

- Does **not** scale well
- Multiple copies sent over the same link
- Central site must know all who participate

Behavior could be changed by explicitly building a tree of reflectors - but then you are moving over to Steve Deering's model.

Maguire
maguire@it.kth.se

Alternative centralized model
2006.03.13

Multicasting and RSVP 421 of 493
Internetworking/Internetteknik

Multicast Backbone (MBONE)

Expanding multicasting across WANs

World-wide, IP-based, real-time conferencing over the Internet (via the MBONE) in daily use for several years with more than 20,000 users in more than 1,500 networks in events carrier to 30 countries.

For a nice paper examining multicast traffic see: "Measurements and Observations of IP Multicast Traffic" by Bruce A. Mah <bmah@CS.Berkeley.EDU>, The Tenet Group, University of California at Berkeley, and International Computer Science Institute, CSD-94-858, 1994,12 pages:

http://www.kitchenlab.org/www/bmah/Papers/Ipmcast-TechReport.pdf/

Maguire
maguire@it.kth.se

Multicast Backbone (MBONE)
2006.03.13

Multicasting and RSVP 422 of 493
Internetworking/Internetteknik

IP Multicast scales well

- End-nodes know nothing about topology
  - Dynamically changes of topology possible, hosts join and leave as they wish
- Routers know nothing about "conversations"
  - changes can be done without global coordination
  - no end-to-end state to move around

Participants view of Multicast
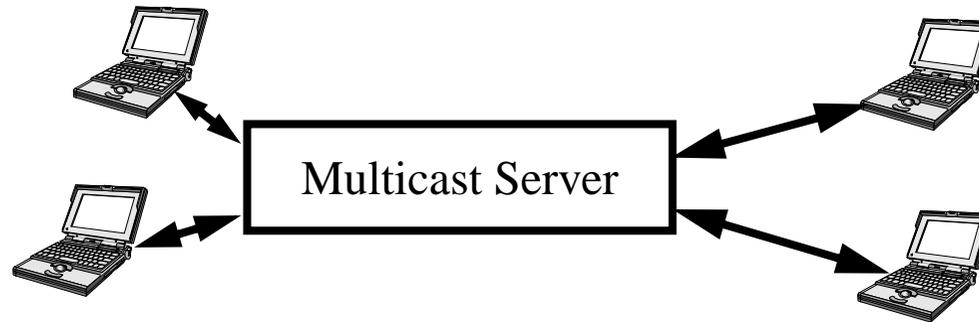
Figure 77: MBONE behaves as if there were a multicast server, but this functionality is distributed not centralized.

Maguire
maguire@it.kth.se

IP Multicast scales well
2006.03.13

Multicasting and RSVP 423 of 493
Internetworking/Internetteknik

Core Problem

How to do efficient multipoint distribution (i.e., at most one copy of a packet crossing any particular link) without exposing topology to end-nodes
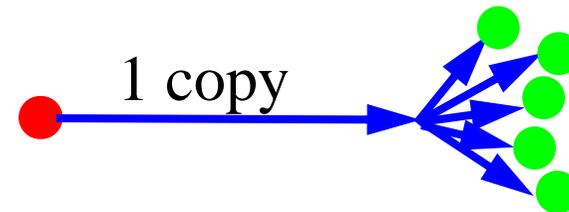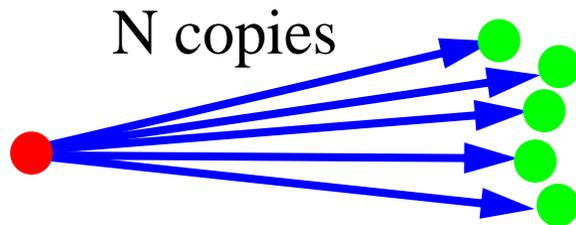
Applications

- Conference calls (without sending N copies sent for N recipients)
- Dissemination of information (stock prices, "radio stations", …)
- Dissemination of one result for many similar requests (boot information, video)
- Unix tools:
  - nv - network video
  - vat - visual audio tool
  - wb - whiteboard
  - sd - session directory
  - …

Maguire
maguire@it.kth.se

Core Problem
2006.03.13

Multicasting and RSVP 424 of 493
Internetworking/Internetteknik

Steve Deering's Multicast

Dynamically constructs efficient delivery trees from sender(s) to receiver(s)

- Key is to compute a spanning tree of multicast routers

Simple service model:

- receivers announce interest in some multicast address
- senders just send to that address
- routers conspire to deliver sender's data to all interested receivers
  - so the real work falls once again to the routers, not the end nodes
  - Note that the assumption here is that it is worth loading the routers with this extra work, because it reduces the traffic which has to be carried.

IGMP v1, v2, v3

Multicast Routing Protocols
PIM, CBT, DVMRP, MOSPF, MBGP, …

Link-level Multicast (Ethernet)
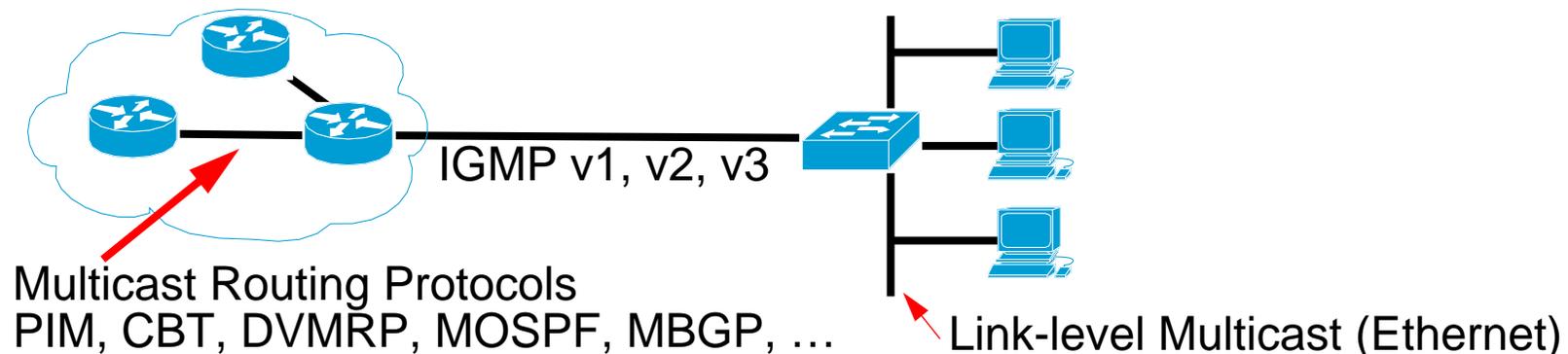
Figure 78: IP Multicast Service Model

Maguire
maguire@it.kth.se

Steve Deering's Multicast
2006.03.13

Multicasting and RSVP 425 of 493
Internetworking/Internetteknik

IP WAN Multicast Requirements

- Convention for recognizing IP multicast
- Convention for mapping IP to LAN address
- Protocol for end nodes to inform their adjacent routers,
- Protocol for routers to inform neighbor routers
- Algorithm to calculate a spanning tree for message flow
- Transmit data packets along this tree

Maguire
maguire@it.kth.se

IP WAN Multicast Requirements
2006.03.13

Multicasting and RSVP 426 of 493
Internetworking/Internetteknik

Multicasting IP addresses

Multicast Group Addresses - "Class D" IP address

- High 4 bits are 0x1110; which corresponds to the range 224.0.0.0 through 239.255.255.255

- host group ≡ set of hosts listening to a given address
  - membership is dynamic - hosts can enter and leave at will
  - no restriction on the number of hosts in a host group
  - a host need not belong in order to send to a given host group
  - permanent host groups - assigned well know addresses by IANA
    - 224.0.0.1 - all systems on this subnet
    - 224.0.0.2 - all routers on this subnet
    - 224.0.0.4 - DVMRP routers
    - 224.0.0.9 - RIP-2 routers
    - 224.0.1.1 - Network Time Protocol (NTP) - see RFC 1305 and RFC 1769 (SNTP)
    - 224.0.1.2 - SGI's dogfight application

Maguire
maguire@it.kth.se

Multicasting IP addresses
2006.03.13

Multicasting and RSVP 427 of 493
Internetworking/Internetteknik

Internet Multicast Addresses

http://www.iana.org/assignments/multicast-addresses listed in DNS under MCAST.NET and 224.IN-ADDR.ARPA.

- 224.0.0.0 - 224.0.0.255 (224.0.0/24) Local Network Control Block
- 224.0.1.0 - 224.0.1.255 (224.0.1/24) Internetwork Control Block
- 224.0.2.0 - 224.0.255.0   AD-HOC Block
- 224.1.0.0 - 224.1.255.255 (224.1/16) ST Multicast Groups
- 224.2.0.0 - 224.2.255.255 (224.2/16) SDP/SAP Block
- 224.3.0.0 - 224.251.255.255 Reserved
- 239.0.0.0/8 Administratively Scoped
    - 239.000.000.000-239.063.255.255 Reserved
    - 239.064.000.000-239.127.255.255 Reserved
    - 239.128.000.000-239.191.255.255 Reserved
    - 239.192.000.000-239.251.255.255 Organization-Local Scope
    - 239.252.0.0/16 Site-Local Scope (reserved)
    - 239.253.0.0/16 Site-Local Scope (reserved)
    - 239.254.0.0/16 Site-Local Scope (reserved)
    - 239.255.0.0/16 Site-Local Scope
    - 239.255.002.002 rasadv

Maguire
maguire@it.kth.se

Internet Multicast Addresses
2006.03.13

Multicasting and RSVP 428 of 493
Internetworking/Internetteknik

Converting Multicast Group to Ethernet Address

Could have been a simple mapping of the 28 bits of multicast group to 28 bits of Ethernet multicast space (which is 2^{27} in size), but this would have meant that IEEE would have to allocate multiple blocks of MAC addresses to this purpose, but:

- they didn't want to allocate multiple blocks to one organization
- a block of 2^{24} addresses costs $1,000 ==> $16K for 2^{27} addresses

Maguire
maguire@it.kth.se

Converting Multicast Group to Ethernet Address
2006.03.13

Multicasting and RSVP 429 of 493
Internetworking/Internetteknik

Mapping Multicast (Class D) address to Ethernet MAC Address

Solution IANA has one block of ethernet addresses 00:00:5e as the high 24 bits

- they decided to give 1/2 this address space to multicast -- thus multicast has the address range: 00:00:5e:00:00:00 to 00:00:5e:7f:ff:ff

- since the first bit of an ethernet multicast has a low order 1 bit (which is the first bit transmitted in link layer order), the addresses are 01:00:5e:00:00:00 to 01:00:5e:7f:ff:ff

- thus there are 23 bits available for use by the 28 bits of the multicast group ID; we just use the bottom 23 bits
    - therefore 32 different multicast group addresses map to the same ethernet address
    - the IP layer will have to sort these 32 out
    - thus although the filtering is not complete, it is very significant

The multicast datagrams are delivered to all processes that belong to the same multicast group.

To extend beyond a single subnet we use IGMP.

Maguire
maguire@it.kth.se

Mapping Multicast (Class D) address to Ethernet MAC Address
2006.03.13

Multicasting and RSVP 430 of
Internetworking/Internetteknik

Problems

Unfortunately many links do not support link layer multicasts at all!

For example:

- ATM
- Frame relay
- many cellular wireless standards
- …

Maguire
maguire@it.kth.se

Problems
2006.03.13

Multicasting and RSVP 431 of 493
Internetworking/Internetteknik

IGMP: Internet Group Management Protocol

IGMP: Internet Group Management Protocol (RFC 1112) [61]:

- Used by hosts and routers to know which hosts currently belong to which multicast groups.

- multicast routers have to know which interface to forward datagrams to

- IGMP like ICMP is part of the IP layer and is transmitted using IP datagrams (protocol = 2) l

IP header	IGMP message
20 bytes	8 bytes
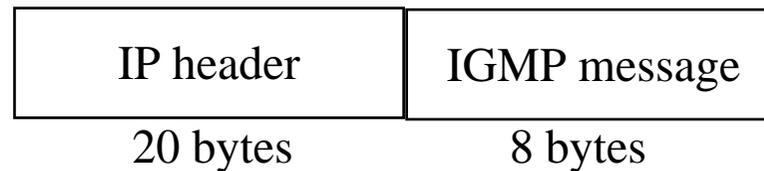
Figure 79: Encapsulation of IGMP message in IP datagram (see Stevens, Vol. 1, figure 13.1, pg. 179)

4 bit IGMP version (1)	4-bit IGMP type (1-2)	Unused	16 bit checksum
		32 bit group address (class D IP address)	
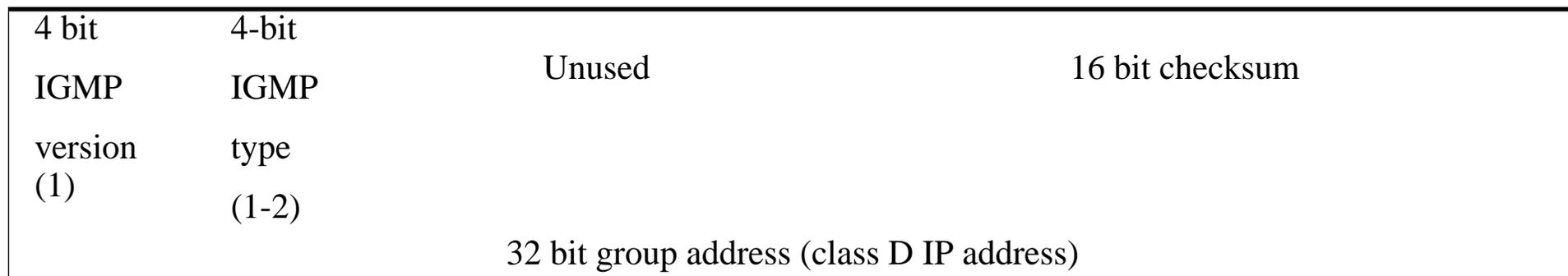
- type =1 $\Rightarrow$ query sent by a router, type =2 $\Rightarrow$ response sent by a host

Maguire
maguire@it.kth.se

IGMP: Internet Group Management Protocol
2006.03.13

Multicasting and RSVP 432 of 493
Internetworking/Internetteknik

How does IGMP fit into the protocol stack
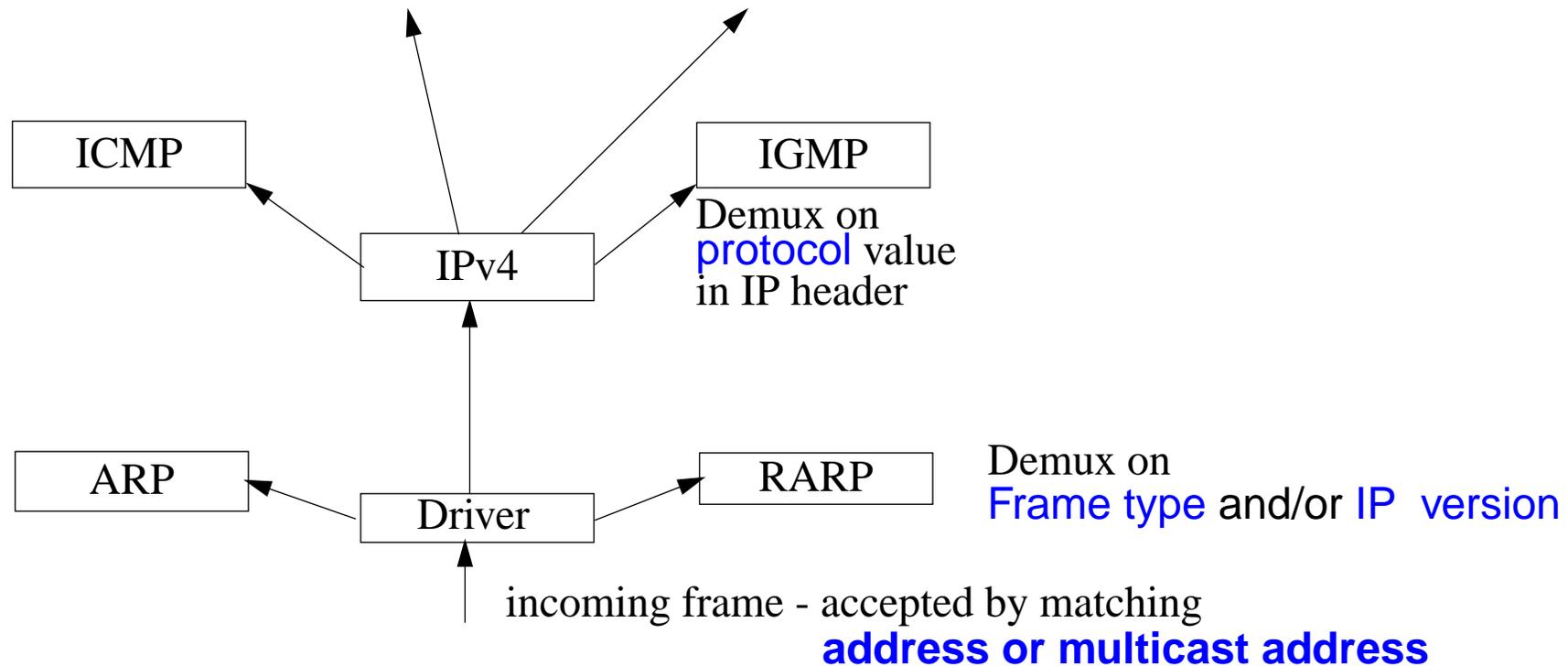
Figure 80: IGMP - adapted from earlier figure (See "Demultiplexing" on page 30.)

So it used IP packets with a protocol value of 2.

Maguire
maguire@it.kth.se

How does IGMP fit into the protocol stack
2006.03.13

Multicasting and RSVP 433 of 493
Internetworking/Internetteknik

Joining a Multicast Group

- a process joins a multicast group on a given interface
- host keeps a table of all groups which have a reference count ≥ 1

IGMP Reports and Queries

- Hosts sends a report when first process joins a given group
- Nothing is sent when processes leave (not even when the last leaves), but the host will no longer send a report for this group
- IGMP router sends queries (to address 224.0.0.1) periodically (one out each interface), the group address in the query is 0.0.0.0

In response to a query, a host sends a IGMP report for every group with at least one process

Routers

- Note that routers have to listen to all 2^{23} link layer multicast addresses!
- Hence they listen promiscuously to all LAN multicast traffic

Maguire
maguire@it.kth.se

Joining a Multicast Group
2006.03.13

Multicasting and RSVP 434 of 493
Internetworking/Internetteknik

IGMP Implementation Details

In order to improve its efficiency there are several clever features:

- Since initial reports could be lost, they are resent after a random time [0, 10 sec]
- Response to queries are also delayed randomly - but if a node hears someone else report membership in a group it is interested in, its response is cancelled

Note: multicast routers don't care which host is a member of which group; only that *someone* attached to the subnet on a given interface is!

Time to Live

- TTL generally set to 1, but you can perform an expanding ring search for a server by increasing the value

- Addresses in the special range 224.0.0.0 through 224.0.0.255 - should never be forwarded by routers - regardless of the TTL value

All-Hosts Group

- all-hosts group address 224.0.0.1 - consists of all multicast capable hosts and routers on a given physical network; membership is *never* reported (sometimes this is called the "all-systems multicast address")

All-Routers Group

- all-routers group address 224.0.0.2

Maguire
maguire@it.kth.se

IGMP Implementation Details
2006.03.13

Multicasting and RSVP 435 of 493
Internetworking/Internetteknik

Group membership State Transitions

adapted from Comer figure 17.4 pg. 330

another host responds/**cancel timer**

join group/**start timer**

timer expires/**send response**

Non-Member

Delaying Member

Member

leave group/**cancel timer**

query arrives/**start timer**

reference count becomes zero/**leave group**

Maguire
maguire@it.kth.se

Group membership State Transitions
2006.03.13

Multicasting and RSVP 436 of 493
Internetworking/Internetteknik
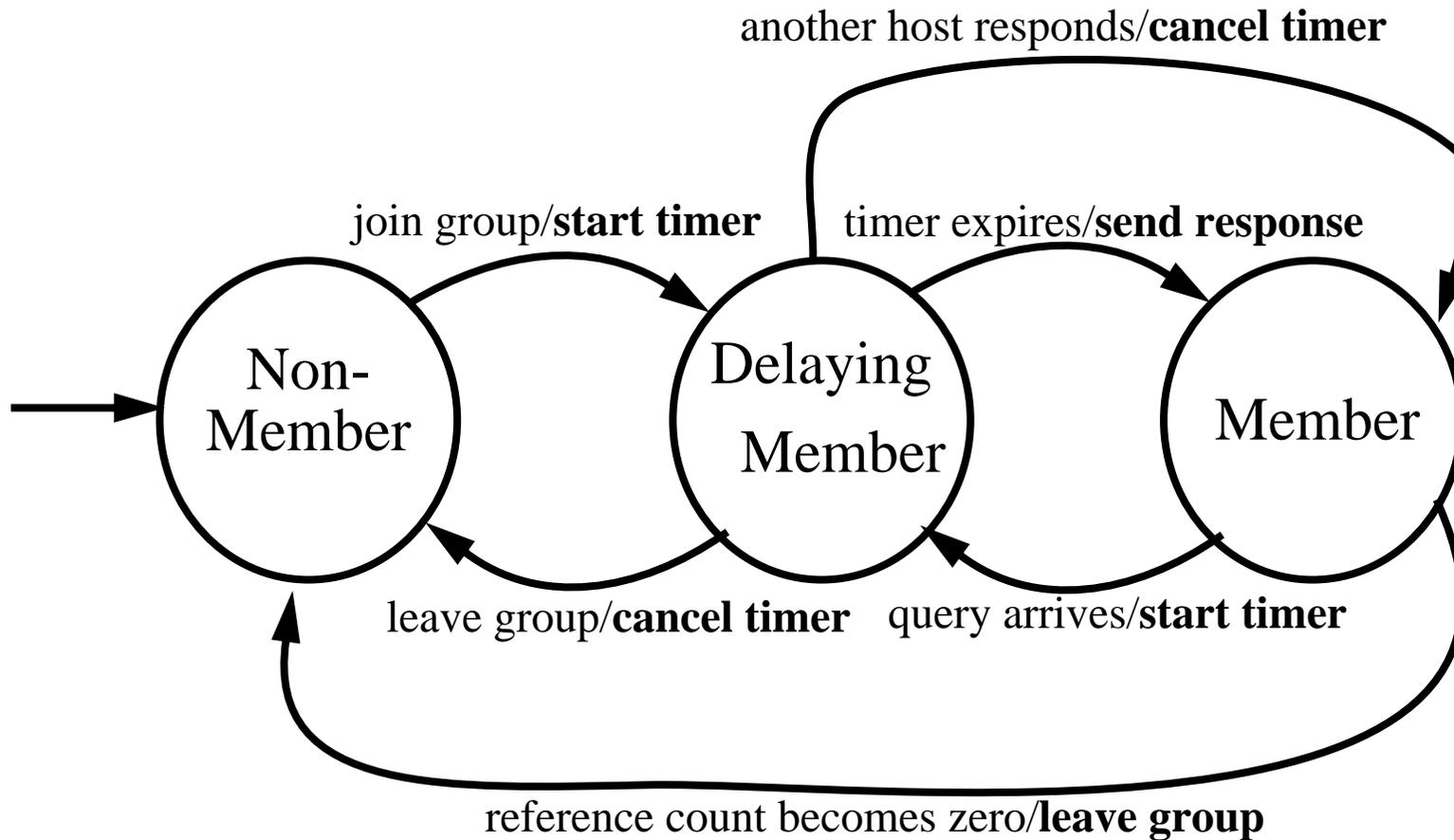
IGMP Version 2 [62]

Allows a host to send a message (to address 224.0.0.2) when they want to explicitly leave a group -- after this message the router sends a *group-specific* query to ask if there is anyone still interested in listening to this group.

- however, the router may have to ask multiple times because this query could be lost

- hence the leave is not immediate -- even if there had been only one member (since the router can't know this)

Maguire
maguire@it.kth.se

IGMP Version 2 [62]
2006.03.13

Multicasting and RSVP 437 of 493
Internetworking/Internetteknik

IGMP Version 3 [63]

- Joining a multicast group, but with a specified set of sender(s) -- so that a client can limit the set of senders which it is interested in hearing from (i.e., source filtering)

- all IGMP replies are now set to a single layer 2 multicast address (e.g., 224.0.0.22) which all IGMPv3-capable multicast routers listen to:
  - because most LANs are now *switched* rather than shared media -- it uses less bandwidth to **not** forward all IGMP replies to all ports
  - most switches now support IGMP snooping -- i.e., the switch is IGMP aware and knows which ports are part of which multicast group (this requires the switch to know which ports other switches and routers are on -- so it can forward IGMP replies to them)
    - switches can listen to this specific layer 2 multicast address - rather than having to listen to all multicast addresses
  - it is thought that rather than have end nodes figure out if all the multicast senders which it is interested in have been replied to - simply make the switch do this work.

Maguire
maguire@it.kth.se

IGMP Version 3 [63]
2006.03.13

Multicasting and RSVP 438 of 493
Internetworking/Internetteknik

IGMP - ethereal

No.	Time	Source	Destination	Protocol	Info
1	0.000000	130.237.15.194	224.0.0.1	IGMP	V2 Membership Query
2	0.632486	130.237.15.225	239.255.255.250	IGMP	V2 Membership Report
3	0.727178	130.237.15.218	239.255.255.250	IGMP	V2 Membership Report
4	1.910951	130.237.15.227	224.0.0.252	IGMP	V2 Membership Report
5	6.953857	130.237.15.229	224.0.1.60	IGMP	V1 Membership Report
6	60.000053	130.237.15.194	224.0.0.1	IGMP	V2 Membership Query
7	61.998827	130.237.15.227	224.0.0.252	IGMP	V2 Membership Report
8	66.711434	130.237.15.225	239.255.255.250	IGMP	V2 Membership Report
9	66.953288	130.237.15.229	224.0.1.60	IGMP	V1 Membership Report
10	120.004228	130.237.15.194	224.0.0.1	IGMP	V2 Membership Query
11	120.872195	130.237.15.218	239.255.255.250	IGMP	V2 Membership Report
12	126.952839	130.237.15.229	224.0.1.60	IGMP	V1 Membership Report
13	129.597716	130.237.15.227	224.0.0.252	IGMP	V2 Membership Report
14	154.655463	211.105.145.186	224.0.0.2	IGMP	V2 Leave Group
15	154.656338	211.105.145.186	224.0.0.2	IGMP	V2 Leave Group
16	180.004408	130.237.15.194	224.0.0.1	IGMP	V2 Membership Query
17	180.943331	130.237.15.217	239.255.255.250	IGMP	V2 Membership Report

⊞ Frame 1 (60 bytes on wire, 60 bytes captured)

Figure 81: IGMP packets as seen with Ethereal

Maguire
maguire@it.kth.se

IGMP - ethereal
2006.03.13

Multicasting and RSVP 439 of 493
Internetworking/Internetteknik

Frame 1: IGMP Membership Query

```
Ethernet II, Src: 00:02:4b:de:ea:d8, Dst: 01:00:5e:00:00:01
      Destination: 01:00:5e:00:00:01 (01:00:5e:00:00:01)
      Source: 00:02:4b:de:ea:d8 (Cisco_de:ea:d8)
      Type: IP (0x0800)
Internet Protocol, Src Addr: 130.237.15.194 (130.237.15.194), Dst
Addr: 224.0.0.1 (224.0.0.1)
      Version: 4
      Header length: 20 bytes
      Differentiated Services Field: 0xc0 (DSCP 0x30: Class Selector
6; ECN: 0x00)
      Total Length: 28
      Identification: 0x6fa3 (28579)
      Flags: 0x00
        Fragment offset: 0
      Time to live: 1
      Protocol: IGMP (0x02)
      Header checksum: 0xd6cc (correct)
      Source: 130.237.15.194 (130.237.15.194)
      Destination: 224.0.0.1 (224.0.0.1)
Internet Group Management Protocol
      IGMP Version: 2
      Type: Membership Query (0x11)
      Max Response Time: 10.0 sec (0x64)
      Header checksum: 0xee9b (correct)
      Multicast Address: 0.0.0.0 (0.0.0.0)
```

Maguire
maguire@it.kth.se

Frame 1: IGMP Membership Query
2006.03.13

Multicasting and RSVP 440 of 493
Internetworking/Internetteknik

Frame 2: IGMP v2 Membership Report

```
        Ethernet II, Src: 00:06:1b:d0:98:c6, Dst: 01:00:5e:7f:ff:fa
        Destination: 01:00:5e:7f:ff:fa (01:00:5e:7f:ff:fa)
        Source: 00:06:1b:d0:98:c6 (Portable_d0:98:c6)
        Type: IP (0x0800)
Internet Protocol, Src Addr: 130.237.15.225 (130.237.15.225), Dst
Addr: 239.255.255.250 (239.255.255.250)
```

Version: 4

Header length: 24 bytes

Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
Total Length: 32
Identification: 0x1f8b (8075)
Flags: 0x00
Time to live: 1

```
        Protocol: IGMP (0x02)
        Header checksum: 0x8284 (correct)
        Source: 130.237.15.225 (130.237.15.225)
        Destination: 239.255.255.250 (239.255.255.250)
        Options: (4 bytes)
              Router Alert: Every router examines packet
Internet Group Management Protocol
        IGMP Version: 2
        Type: Membership Report (0x16)
        Max Response Time: 0.0 sec (0x00)
        Header checksum: 0xfa04 (correct)
        Multicast Address: 239.255.255.250 (239.255.255.250)
```

Maguire
maguire@it.kth.se

Frame 2: IGMP v2 Membership Report
2006.03.13

Multicasting and RSVP 441 of 493
Internetworking/Internetteknik

Frame 12: IGMP v1 Membership Report

```
Ethernet II, Src: 00:01:e6:a7:d3:b9, Dst: 01:00:5e:00:01:3c
     Destination: 01:00:5e:00:01:3c (01:00:5e:00:01:3c)
     Source: 00:01:e6:a7:d3:b9 (Hewlett-_a7:d3:b9)
     Type: IP (0x0800)
Internet Protocol, Src Addr: 130.237.15.229 (130.237.15.229), Dst
Addr: 224.0.1.60 (224.0.1.60)
     Version: 4
     Header length: 20 bytes
     Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN:
0x00)
     Total Length: 28
     Identification: 0x01f6 (502)
     Flags: 0x00
     Fragment offset: 0
     Time to live: 1
     Protocol: IGMP (0x02)
     Header checksum: 0x43dc (correct)
     Source: 130.237.15.229 (130.237.15.229)
     Destination: 224.0.1.60 (224.0.1.60)
Internet Group Management Protocol
     IGMP Version: 1
     Type: Membership Report (0x12)
     Header checksum: 0x0cc3 (correct)
     Multicast Address: 224.0.1.60 (224.0.1.60)
```

Maguire
maguire@it.kth.se

Frame 12: IGMP v1 Membership Report
2006.03.13

Multicasting and RSVP 442 of 493
Internetworking/Internetteknik

Frame 15: IGMP v2 Leave Group

```
Ethernet II, Src: 00:02:8a:78:91:8f, Dst: 01:00:5e:00:00:02
      Destination: 01:00:5e:00:00:02 (01:00:5e:00:00:02)
      Source: 00:02:8a:78:91:8f (AmbitMic_78:91:8f)
      Type: IP (0x0800)
Internet Protocol, Src Addr: 211.105.145.186 (211.105.145.186), Dst
Addr: 224.0.0.2 (224.0.0.2)
      Version: 4
      Header length: 24 bytes
      Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
      Total Length: 32
      Identification: 0x9391 (37777)
      Flags: 0x00
      Fragment offset: 0
      Time to live: 1
      Protocol: IGMP (0x02)
      Header checksum: 0x4c20 (correct)
      Source: 211.105.145.186 (211.105.145.186)
      Destination: 224.0.0.2 (224.0.0.2)
      Options: (4 bytes)
            Router Alert: Every router examines packet
Internet Group Management Protocol
      IGMP Version: 2
      Type: Leave Group (0x17)
      Max Response Time: 0.0 sec (0x00)
      Header checksum: 0xff71 (correct)
      Multicast Address: 239.192.249.204 (239.192.249.204)
```

Maguire
maguire@it.kth.se

Frame 15: IGMP v2 Leave Group
2006.03.13

Multicasting and RSVP 443 of 493
Internetworking/Internetteknik
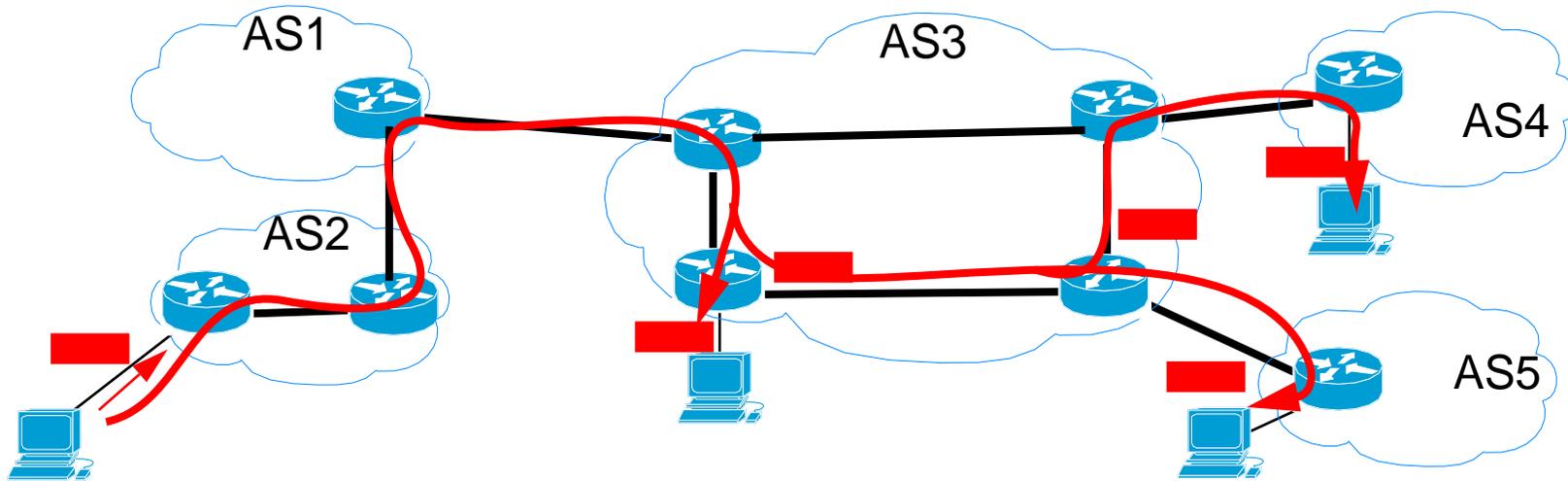
Multicast routing

Figure 82: Multicast routing: packet replicated by the routers -- not the hosts

- packet forwarded one or more interfaces
  - router replicates the packet as necessary
- need to build a delivery tree - to decide on which paths to forward

Maguire
maguire@it.kth.se

Multicast routing
2006.03.13

Multicasting and RSVP 444 of 493
Internetworking/Internetteknik

Therefore a Multicast Router

- ## Listens to all multicast traffic and forwards *if* necessary
  - Listens promiscuously to all LAN multicast traffic

- ## Listens to all multicast addresses
  - For an ethernet this means all 2^{23} link layer multicast addresses

- ## Communicates with:
  - directly connected hosts via IGMP
  - other multicast routers with multicast routing protocols

Maguire
maguire@it.kth.se

Multicast routing
2006.03.13

Multicasting and RSVP 445 of 493
Internetworking/Internetteknik

Multicasting

Example: Transmitting a file from C to A, B, and D.

✘ Using point-to-point transfer, some links will be used more than once to send the same file
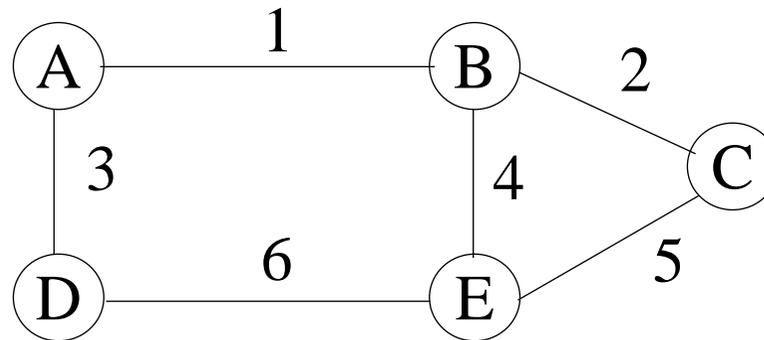
✔ Using Multicast

Point-to-point						
Link	A	B	E	D	Total	Multicast
1	1				1	1
2	1	1			2	1
5			1	1	2	1
6				1	1	1
	2	1	1	2		4

Maguire
maguire@it.kth.se

Multicast routing
2006.03.13

Multicasting and RSVP 446 of 493
Internetworking/Internetteknik

Multicast Routing - Flooding

- maintaining a list of recently seen packets (last 2 minutes), if it has been seen before, then delete it, otherwise copy to a cache/database and send a copy on all (except the incoming) interface.

  ✘ Disadvantages:

  - ◆ Maintaining a list of "last-seen" packets. This list can be fairly long in high speed networks
  - ◆ The "last-seen" lists guarantee that a router will not forward the same packet twice, but it certainly does not guarantee that the router will receive a packet only once.

  ✔ Advantages

  - ◆ Robustness
  - ◆ It does not depend on any routing tables.

Maguire
maguire@it.kth.se
Multicast Routing - Flooding
2006.03.13
Multicasting and RSVP 447 of 493
Internetworking/Internetteknik

Delivery Trees: different methods

- ## Source-based Trees
  - Notation: (S, G) $\Rightarrow$ only specific sender(s) [S= source, G=Group]
  - Uses memory proportional to O(S*G), can find optimal paths $\Rightarrow$ minimizes delay

- ## Group Shared Trees
  - Notation: (*, G) $\Rightarrow$ All senders
  - Uses less memory (O(G)), but uses suboptimal paths $\Rightarrow$ higher delay

- ## Data-driven
  - Build only when data packets are sent

- ## Demand-driven
  - Build the tree as members join
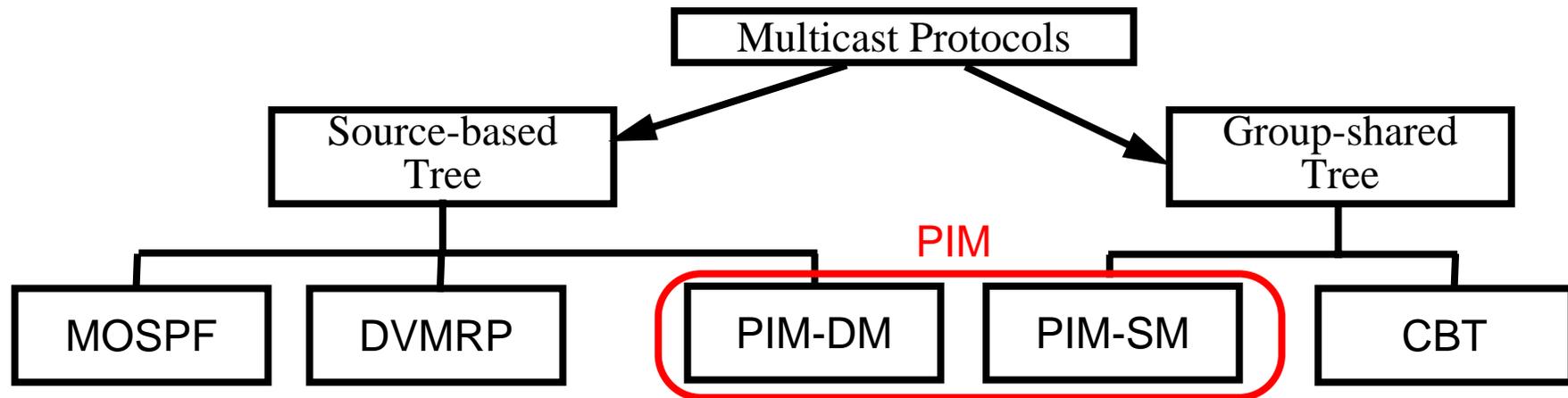
Figure 83: Taxonomy of Multicast Routing Protocols (see Forouzan figure 15.7 pg. 444)

Maguire
maguire@it.kth.se

Delivery Trees: different methods
2006.03.13

Multicasting and RSVP 448 of 493
Internetworking/Internetteknik

Multicast Routing - Spanning Trees

The "spanning tree" technique is used by "media-access-control (MAC) bridges".

- Simply build up an "overlay" network by marking some links as "part of the tree" and other links as "unused" (produces a loopless graph).
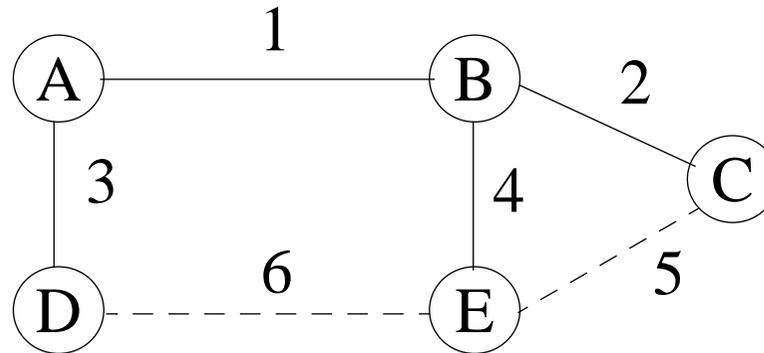
Drawbacks

✗ It does not take into account group membership

✗ It concentrates all traffic into a small subset of the network links.

Maguire
maguire@it.kth.se

Multicast Routing - Spanning Trees
2006.03.13

Multicasting and RSVP 449 of 493
Internetworking/Internetteknik

Link-State Multicast: MOSPF [64]

Just add multicast to a link-state routing protocol thus OSPF $\Rightarrow$ MOSPF

- Use the multiprotocol facility in OSPF to carry multicast information

- Extended with a group-membership LSA
  - This LSA lists only members of a given group

- Use the resulting link-state database to build delivery trees
  - Compute least-cost source-based trees considering metrics using Dijkstra's algorithm
  - A tree is computed for each (S,G) pair with a given source (S), this is done for all S
  - Remember that as a link-state routing protocol that every router will know the topology of the complete network

- However, it is expensive to keep store all this information (and most is unnecessary)
  - Cache only the active (S,G) pairs
  - Use a data-driven approach, i.e., only computes a new tree when a multicast datagram arrives for this group

Maguire
maguire@it.kth.se

Link-State Multicast: MOSPF [64]
2006.03.13

Multicasting and RSVP 450 of 493
Internetworking/Internetteknik

Reverse -Path Forwarding (RPF)

RPF algorithm takes advantage of a routing table to "orientate" the network and to compute an implicit tree per network source.

Procedure

1. When a multicast packet is received, note source (S) and interface (I)

2. If I belongs to the shortest path toward S, forward to all interfaces except I.

- Compute shortest path **from** the **source** to the node rather than from the node to the source.

- Check whether the local router is on the shortest path between a neighbor and the source before forwarding a packet to that neighbor. If this is not the case, then there is no point in forwarding a packet that will be immediately dropped by the next router.

Maguire
maguire@it.kth.se

Reverse -Path Forwarding (RPF)
2006.03.13

Multicasting and RSVP 451 of 493
Internetworking/Internetteknik

- RPF results in a different spanning tree for each source.
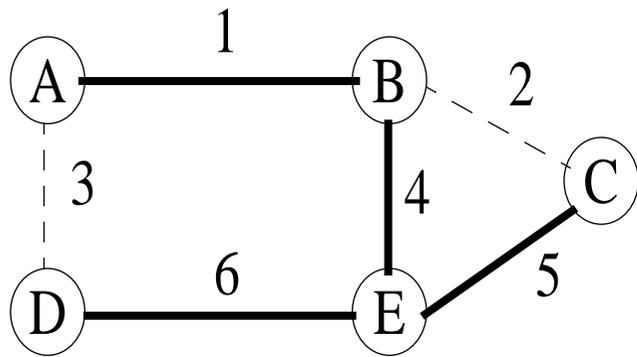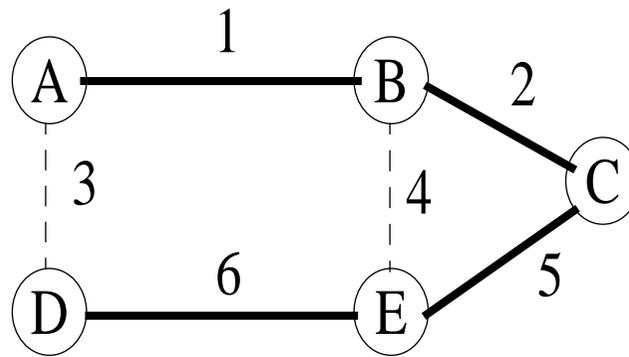
RPF tree from E   RPF tree from C   RPF tree from A
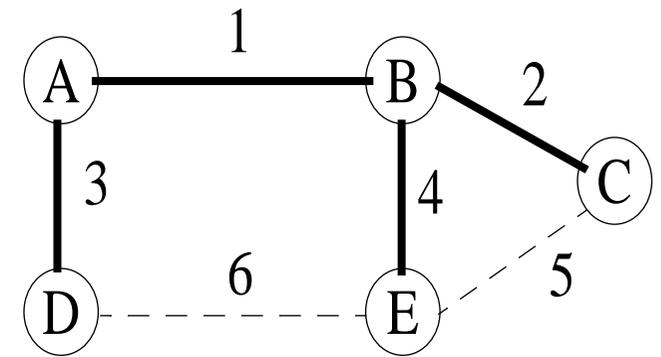
These trees have two interesting properties:

- They guarantee the fastest possible delivery, as multicasting follows the shortest path from source to destination
- Better network utilization, since the packets are spread over multiple links.

Drawback

✘ Group membership is **not** taken into account when building the tree
⇒ a network can receive two or more copies of a multicast packet

Reverse Path Broadcast (RPB)

- We define a parent router for each network

- For each source, a router will forward a multicast packet **only** if it is the designated parent

$\Rightarrow$ each network gets only one copy of each multicast packet

Maguire
maguire@it.kth.se

Reverse Path Broadcast (RPB)
2006.03.13

Multicasting and RSVP 453 of 493
Internetworking/Internetteknik

RPB + Prunes ⇒ Reverse Path Multicast (RPM)

When source S starts a multicast transmission the first packet is propagated to all the network nodes (i.e., flooding). Therefore all leaf nodes receive the first multicast packet. However, if there is a leaf node that does **not** want to receive further packets, it will send back a "prune" message to the router that sent it this packet - saying effectively "don't send further packets from source S to group G on this interface I."

There are two obvious drawback in the flood and prune algorithm:

- The first packet is flooded to the whole network
- The routers must keep states per group and source

When a listener joins at a leaf that was pruned, we add this leaf back by grafting.

Flood and prune was acceptable in the experimental MBONE which had only a few tens of thousands of nodes, but for the Internet where both the number of sources and the number of groups becomes very large, there is a risk of exhausting the memory resources in network routers.

Maguire
maguire@it.kth.se

RPB + Prunes ⇒ Reverse Path Multicast (RPM)
2006.03.13

Multicasting and RSVP 454 of 493
Internetworking/Internetteknik

Distance-Vector Multicast Routing Protocol (DVMRP) [65]

- Start with a unicast distance-vector routing protocol (e.g., RIP), then extend (Destination, Cost, Nexthop) $\Rightarrow$ (Group, Cost, Nexthops)
  - Routers only know their next hop (i.e., which neighbor)
- Reverse Path Multicasting (RPM)
- DVMRP is data-driven and uses source-based trees

Maguire
maguire@it.kth.se
Distance-Vector Multicast Routing Protocol (DVMRP) [65] Multicasting and RSVP 455 of 493
2006.03.13
Internetworking/Internetteknik

Multicast Routing - Steiner Tree's

Assume source C and the recipients are A and D.
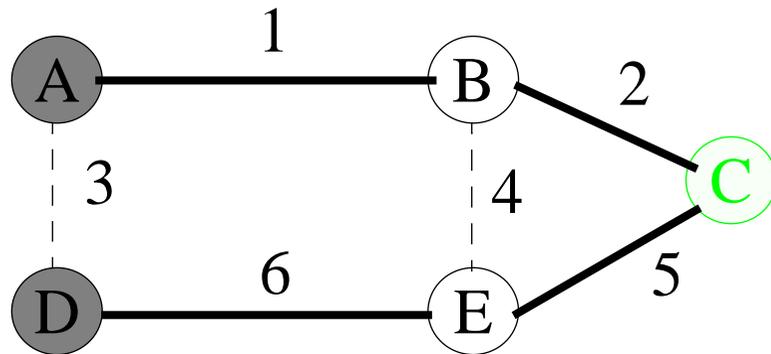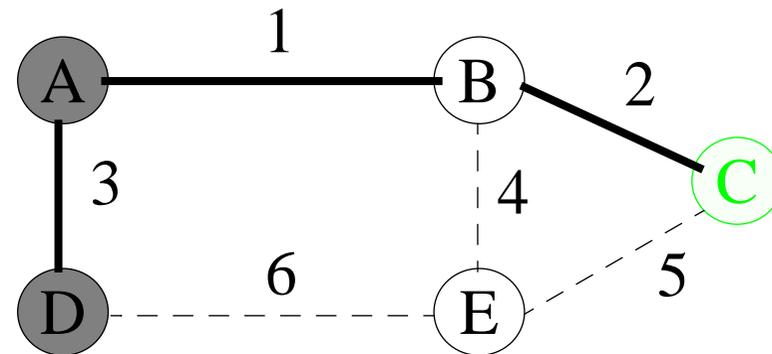
RPF Tree (4 links)   Seiner Tree (3 links)

Figure 84: RPF vs. Steiner Tree

- Steiner tree uses less resources (links), but are *very hard* to compute (N-P complete)

- In Steiner trees the routing changes widely if a new member joins the group, this leads to instability. Thus the Steiner tree is more a mathematical construct that a practical tool.

Maguire
maguire@it.kth.se

Multicast Routing - Steiner Tree's
2006.03.13

Multicasting and RSVP 456 of 493
Internetworking/Internetteknik

Core-Based Trees (CBT)

A fixed point in the network chosen to be the center of the multicast group, i.e., "core". Nodes desiring to be recipients send "join" commands toward this core. These commands will be processed by all intermediate routers, which will mark the interface on which they received the command as belonging to the group's tree. The routers need to keep one piece of state information per group, listing all the interface that belong to the tree. If the router that receives a join command is already a member of the tree, it will mark only one more interface as belong to the group. If this is the first join command that the router receives, it will forward the command one step further toward the core.

Advantages

- CBT limits the expansion of multicast transmissions to precisely the set of all recipients (so it is demand-driven). This is in contrast with RPF where the first packet is sent to the whole network.

- The amount of state is less; it depends only on the number of the groups, not the number of pairs of sources and groups $\Rightarrow$ Group-shared multicast trees (*, G)

- Routing is based on a spanning tree, thus CBT does **not** depend on multicast or unicast routing tables

Disadvantages

- The path between some sources and some receivers may be suboptimal.

- Senders sends multicast datagrams to the core router encapsulated in unicast datagrams

Maguire
maguire@it.kth.se

Core-Based Trees (CBT)
2006.03.13

Multicasting and RSVP 457 of 493
Internetworking/Internetteknik

Protocol-Independent Multicast (PIM)

Two modes:

- ## PIM-dense mode (PIM-DM) [67]
    - Dense mode is an implementation of RPF and prune/graft strategy
    - Relies on unicast routing tables providing an optimal path
    - However, it is independent of the underlying unicast protocol

- ## PIM-sparse mode (PIM-SM) [66]
    - Sparse mode is an implementation of CBT where join points are called "rendezvous points"
    - A given router may know of more than one rendezvous point
    - Simpler than CBT as there is no need for acknowledgement of a join message
    - Can switch from group-shared tree to source-based tree if there is a dense cluster far from the nearest rendezvous point

The adjectives "dense" and "sparse: refer to the density of group members in the Internet. Where a group is send to be **dense** if the probability is high that the area contains at least one group member. It is send to be **sparse** if that probability is low.

Maguire
maguire@it.kth.se
Protocol-Independent Multicast (PIM)
2006.03.13
Multicasting and RSVP 458 of 493
Internetworking/Internetteknik

Multiprotocol BGP (MBGP) [69]

Extends BGP to enable multicast routing policy, thus it connects multicast topologies within and between BGP autonomous systems

Add two new (optional and non-transitive) attributes:

- Multiprotocol Reachable NLRI (MP_REACH_NLRI)
- Multiprotocol Unreachable NLRI (MP_UNREACH_NLRI)

As these are optional and non-transitive attributes - routers which do not support these attributes ignore then and don't pass them on.

Thus MBGP allows the exchange of multicast routing information, but one must still use PIM to build the distribution tree to actually forward the traffic!

Maguire
maguire@it.kth.se
Multiprotocol BGP (MBGP) [69]
2006.03.13
Multicasting and RSVP 459 of 493
Internetworking/Internetteknik

Multicast backbone (MBONE) [61]

Why can you do when all router's and networks don't support multicasting:
Tunnel!
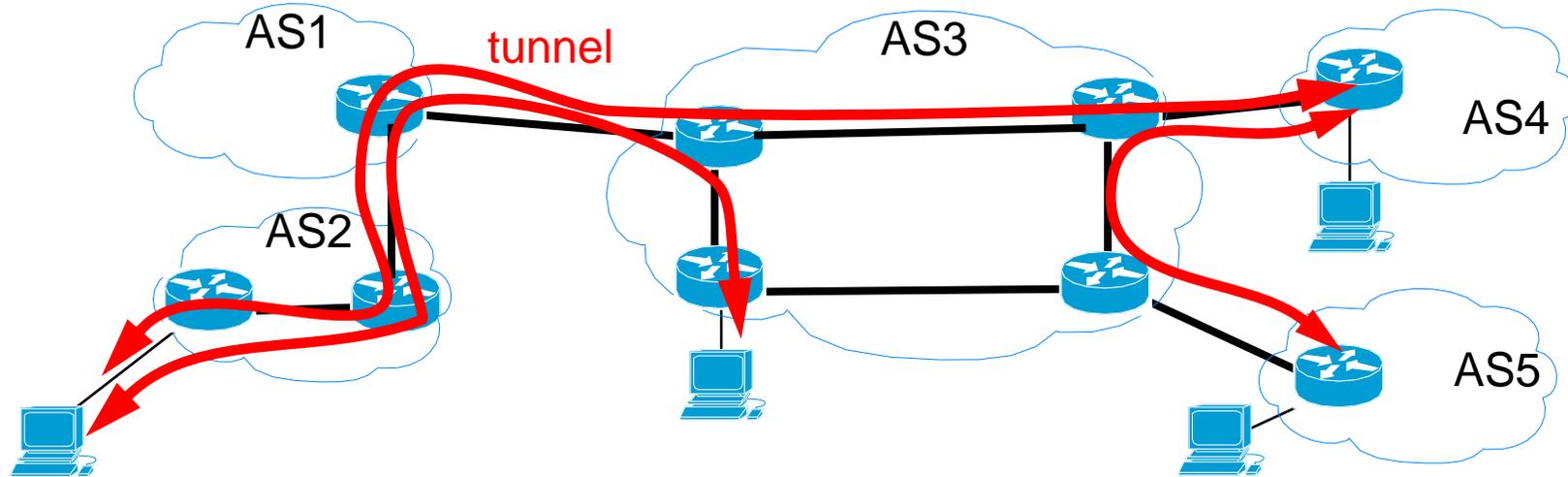
Figure 85: Multicast routing via tunnels - the basis of MBONE (see Forouzan figure 15.14 pg. 453)

See the IETF MBONE Deployment Working Group (MBONED)

http://antc.uoregon.edu/MBONED/ and their charter *http://www.ietf.org/html.charters/mboned-charter.html*

Maguire
maguire@it.kth.se

Multicast backbone (MBONE) [61]
2006.03.13

Multicasting and RSVP 460 of 493
Internetworking/Internetteknik

Telesys class was multicast over MBONE

Already in Period 2, 1994/1995 "Telesys, gk" was multicast over the internet and to several sites in and near Stockholm.

Established ports for each of the data streams:

- electronic whiteboard
- video stream
- audio stream

The technology works - but it is very important to get the audio packets delivered with modest delay and loss rate. Poor audio quality is perceived a major problem.

NASA and several other organizations regularly multicast their audio and video "programs".

Maguire
maguire@it.kth.se

Telesys class was multicast over MBONE
2006.03.13

Multicasting and RSVP 461 of 493
Internetworking/Internetteknik

Benefits for Conferencing

- IP Multicast is efficient, simple, robust
- Users can join a conference without enumerating (or even knowing) other participants
- User can join and leave at any time
- Dynamic membership

Maguire
maguire@it.kth.se

Benefits for Conferencing
2006.03.13

Multicasting and RSVP 462 of 493
Internetworking/Internetteknik

MBONE Chronology

Nov. 1988	Small group proposes testbed net to DARPA. This becomes DARTNET
Nov. 1990	Routers and T1 lines start to work
Feb. 1991	First packet audio conference (using ISI's vt)
Apr. 1991	First multicast audio conference
Sept. 1991	First audio+video conference (hardware codec)
Mar. 1992	Deering & Casner broadcast San Diego IETF to 32 sites in 4 countries
Dec. 1992	Washington DC IETF - four channels of audio and video to 195 watchers in 12 countries
Jan. 1993	MBONE events go from one every 4 months to several a day
1994/1995	Telesys gk -- multicast from KTH/IT in Stockholm
July 1995	KTH/IT uses MBONE to multicast two parallel sessions from IETF meeting in Stockholm
...	
today	lots of users and "multicasters"

IETF meetings are *now* regularily multicast - so the number of participants that can attend is not limited by physical space or travel budgets.

Maguire
maguire@it.kth.se

MBONE Chronology
2006.03.13

Multicasting and RSVP 463 of 493
Internetworking/Internetteknik
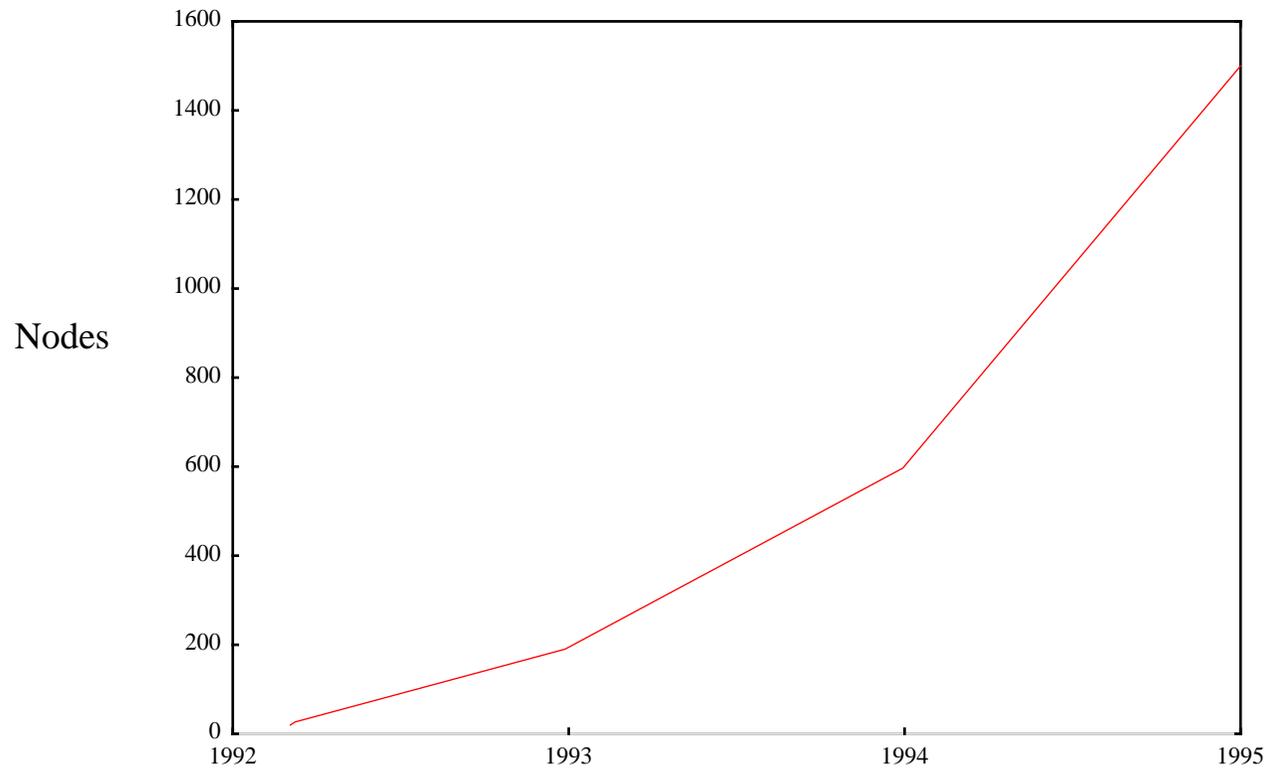
MBONE growth

Figure 86: MBONE Growth - Doubling time ~8 months

Maguire
maguire@it.kth.se

MBONE growth
2006.03.13

Multicasting and RSVP 464 of 493
Internetworking/Internetteknik

For the some statistics see: _http://www.caida.org/tools/measurement/mantra/_

Multicast	2003	2002	Old state 2000
	02/06/2003,15:25:38 PST	01/21/2002,11:30 PST (Pacific Standard Time)	
Entity	Value		
#Groups	4473	1002	330
#Participants	6059	average 4	
#Unique Participants	1446		
#ASes	137		
#RPs	197		

But we are still waiting for multicast to "take off".

Maguire
maguire@it.kth.se

MBONE growth
2006.03.13

Multicasting and RSVP 465 of 493
Internetworking/Internetteknik

MBONE connections

MBONE is an "overlay" on the Internet

- multicast routers were distinct from normal, unicast routers - but increasingly routers support multicasting
- it is not trivial to get hooked up
- requires cooperation from local and regional people

MBONE is changing:

- Most router vendors now support IP multicast
- MBONE will go away as a distinct entity once ubiquitous multicast is supported throughout the Internet.
- Anyone hooked up to the Internet can participate in conferences

Maguire
maguire@it.kth.se

MBONE connections
2006.03.13

Multicasting and RSVP 466 of 493
Internetworking/Internetteknik

mrouted

mrouted UNIX deamon

tunneling to other MBONE routers

See: "Linux-Mrouted-MiniHOWTO: How to set up Linux for multicast routing"
by Bart Trojanowski <bart@jukie.net>, v0.1, 30 October 1999

http://jukie.net/~bart/multicast/Linux-Mrouted-MiniHOWTO.html

and *http://www.linuxdoc.org/HOWTO/Multicast-HOWTO-5.html*

Maguire
maguire@it.kth.se

mrouted
2006.03.13

Multicasting and RSVP 467 of 493
Internetworking/Internetteknik

Multicast Source Discovery Protocol (MSDP)[72]

As the routing protocols deployed in the multicast networks operating in sparse mode do not support flooding information, a mechanism was needed to propagate information about sources (i.e., hosts sourcing data to a multicast group) and the associated multicast groups to all the multicast networks.

Sends Source Active (SA) messages containing (S,G,RP):

- Source Address,
- Group Address,
- and RP Address

these are propagated by Rendezvous Points over TCP

MSDP connects multiple PIM-SM domains together. Each domain uses its own **independent** Rendezvous Point (RP) and does not depend on RPs in other domains.

Maguire
maguire@it.kth.se

Multicast Source Discovery Protocol (MSDP)[72]
2006.03.13

Multicasting and RSVP 468 of 493
Internetworking/Internetteknik

GLOP addressing

Traditionally multicast address allocation has been dynamic and done with the help of applications like SDR that use Session Announcement Protocol (SAP).

GLOP is an example of a policy for allocating multicast addresses (it is still experimental in nature). It allocated the 233/8 range of multicast addresses amongst different ASes such that each AS is statically allocated a /24 block of multicast addresses. See [68]

0	7 8	23	31
233	16 bits AS		local bits

Maguire
maguire@it.kth.se

GLOP addressing
2006.03.13

Multicasting and RSVP 469 of 493
Internetworking/Internetteknik

Single Source Multicast (SSM) [74]

- A single source multicast-address space was allocated to 232/8
- Each AS is allocated a unique 232/24 address block that it can use for multicasting.

Maguire
maguire@it.kth.se

Single Source Multicast (SSM) [74]
2006.03.13

Multicasting and RSVP 470 of 493
Internetworking/Internetteknik

Other multicast efforts

PGM: Pragmatic General Multicast Protocol [73]

Administratively Scoped IP Multicast [75]

…

Maguire
maguire@it.kth.se

Other multicast efforts
2006.03.13

Multicasting and RSVP 471 of 493
Internetworking/Internetteknik

Tools for managing multicast

"Managing IP Multicast Traffic" A White Paper from the IP Multicast Initiative (IPMI) and Stardust Forums for the benefit of attendees of the 3rd Annual IP Multicast Summit, February 7-9, 1999

http://techsup.vcon.com/whtpprs/Managing%20IP%20Multicast%20Traffic.pdf

Mrinfo	shows the multicast tunnels and routes for a router/mrouted.
Mtrace	traces the multicast path between two hosts.
RTPmon	displays receiver loss collected from RTCP messages.
Mhealth	monitors tree topology and loss statistics.
Multimon	monitors multicast traffic on a local area network.
Mlisten	captures multicast group membership information.
Dr. Watson	collects information about protocol operation.

Mantra (Monitor and Analysis of Traffic in Multicast Routers)

http://www.caida.org/tools/measurement/mantra/

Maguire
maguire@it.kth.se

Tools for managing multicast
2006.03.13

Multicasting and RSVP 472 of 493
Internetworking/Internetteknik

SNMP-based tools and multicast related MIBs

Management Information Bases (MIBs) for multicast:

RTP MIB	designed to be used by either host running RTP applications or intermediate systems acting as RTP monitors; has tables for each type of user; collect statistical data about RTP sessions.
Basic Multicast Routing MIB	includes only general data about multicast routing. such as multicast group and source pairs; next hop routing state, forwarding state for each of a router's interfaces, and information about multicast routing boundaries.

Maguire
maguire@it.kth.se

SNMP-based tools and multicast related MIBs
2006.03.13

Multicasting and RSVP 473 of 493
Internetworking/Internetteknik

Protocol-Specific Multicast Routing MIBs

Provide information specific to a particular routing protocol

PIM MIB	list of PIM interfaces that are configured; the router's PIM neighbors; the set of rendezvous points and an association for the multicast address prefixes; the list of groups for which this particular router should advertise itself as the candidate rendezvous point; the reverse path table for active multicast groups; and component table with an entry per domain that the router is connected to.
CBT MIB:	configuration of the router including interface configuration; router statistics for multicast groups; state about the set of group cores, either generated by automatic bootstrapping or by static mappings; and configuration information for border routers.
DVMRP MIB	interface configuration and statistics; peer router configuration states and statistics; the state of the DVMRP (Distance-Vector Multicast Routing Protocol) routing table; and information about key management for DVMRP routes.
Tunnel MIB	lists tunnels that might be supported by a router or host. The table supports tunnel types including Generic Routing Encapsulation (GRE) tunnels, IP-in-IP tunnels, minimal encapsulation tunnels, layer two tunnels (LTTP), and point-to-point tunnels (PPTP).
IGMP MIB	only deals with determining if packets should be forwarded over a particular leaf router interface; contains information about the set of router interfaces that are listening for IGMP messages, and a table with information about which interfaces currently have members listening to particular multicast groups.

Maguire
maguire@it.kth.se

Protocol-Specific Multicast Routing MIBs
2006.03.13

Multicasting and RSVP 474 of 493
Internetworking/Internetteknik

SNMP tools for working with multicast MIBs

Merit SNMP-Based Management Project has release two freeware tools which work with multicast MIBs:

Mstat	queries a router or SNMP-capable mrouted to generate various tables of information including routing tables, interface configurations, cache contents, etc.
Mview	"application for visualizing and managing the MBone",allows user to display and interact with the topology, collect and monitor performance statistics on routers and links

HP Laboratories researchers investigating IP multicast network management are building a prototype integrated with HP OpenView -- intended for use by the network operators who are not experts in IP multicast; provides discovery, monitoring and fault detection capabilities.

Maguire
maguire@it.kth.se

SNMP tools for working with multicast MIBs
2006.03.13

Multicasting and RSVP 475 of 493
Internetworking/Internetteknik

QoS & Scheduling algorithms

Predictable delay is thought to be required for interactive real-time applications:
Alternatives:

1. use a network which guarantees fixed delays

2. use a packet scheduling algorithm

3. retime traffic at destination

Since queueing at routers, hosts, etc. has traditionally been simply FIFO; which does not provide guaranteed end-to-end delay both the 2nd and 3rd method use alternative algorithms to maintain a predictable delay.

Algorithms such as: Weighted Fair Queueing (WFQ)

These algorithms normally emulate a fluid flow model.

As it is very hard to provide fixed delays in a network, hence we will examine the 2nd and 3rd methods.

Maguire
maguire@it.kth.se

QoS & Scheduling algorithms
2006.03.13

Multicasting and RSVP 476 of 493
Internetworking/Internetteknik

RSVP: Resource Reservation Setup Protocol [78]

- RSVP is a network control protocol that will deal with resource reservations for certain Internet applications.

- RSVP is a component of "Integrated services" Internet, and can provide both best-effort and QoS.
  - Applications request a specific quality of service for a data stream

- RSVP delivers QoS requests to each router along the path.
  - Maintains router and host state along the data stream during the requested service.
  - Hosts and routers deliver these request along the path(s) of the data stream
  - At each node along the path RSVP passes a new resource reservation request to an admission control routine

RSVP is a signalling protocol carrying no application data
  - First a host sends IGMP messages to join a group
  - Second a host invokes RSVP to reserve QoS

Maguire
maguire@it.kth.se

RSVP: Resource Reservation Setup Protocol [78]
2006.03.13

Multicasting and RSVP 477 of 493
Internetworking/Internetteknik

Functionality

- RSVP is receiver oriented protocol.
  The receiver is responsible for requesting reservations.

- RSVP handles heterogeneous receivers.
  Hosts in the same multicast tree may have different capabilities and hence need different QoS.

- RSVP adapts to changing group membership and changing routes.
  RSVP maintains "Soft state" in routers. The only permanent state is in the end systems. Each end system sends their RSVP control messages to refresh the router state.
  In the absence of refresh message, RSVP state in the routers will time-out and be deleted.

- RSVP is **not** a routing protocol.
  A host sends IGMP messages to join a multicast group, but it uses RSVP to reserve resources along the delivery path(s) from that group.

Maguire
maguire@it.kth.se

Functionality
2006.03.13

Multicasting and RSVP 478 of 493
Internetworking/Internetteknik

Resource Reservation

- Interarrival variance reduction / jitter
- Capacity assignment / admission control
- Resource allocation (who gets the bandwidth?)

Maguire
maguire@it.kth.se

Resource Reservation
2006.03.13

Multicasting and RSVP 479 of 493
Internetworking/Internetteknik

Jitter Control

- if network has enough capacity
  average departure rate = receiver arrival rate
- Then jitter is caused by queue waits due to competing traffic
- Queue waits should be at most the amount of competing traffic in transit, total amount of in transit data should be at most round trip propagation time
  (100 ms for transcontinental path)
  (64 kbit/sec => buffer = 8 kb/s*0.1 sec = 800 bytes)

See: Jonathan Rosenberg, Lili Qiu, and Henning Schulzrinne, "Integrating Packet FEC into Adaptive Voice Playout Buffer Algorithms on the Internet",INFOCOM, (3), 2000, pp. 1705-1714.

See also *http://citeseer.nj.nec.com/rosenberg00integrating.html*

Maguire
maguire@it.kth.se

Jitter Control
2006.03.13

Multicasting and RSVP 480 of 493
Internetworking/Internetteknik

Capacity Assignment

- end-nodes ask network for bandwidth.
- Can get "yes" or "no" (busy signal)
- Used to control available transmission capacity

Maguire
maguire@it.kth.se

Capacity Assignment
2006.03.13

Multicasting and RSVP 481 of 493
Internetworking/Internetteknik

RSVP Protocol Mechanism

- Sender sends RSVP PATH message which records path
- Receiver sends RSVP RESV message backwards along the path indicating desired QoS
- In case of failure a RSVP error message is returned

Figure 87:

Maguire
maguire@it.kth.se

RSVP Protocol Mechanism
2006.03.13

Multicasting and RSVP 482 of 493
Internetworking/Internetteknik
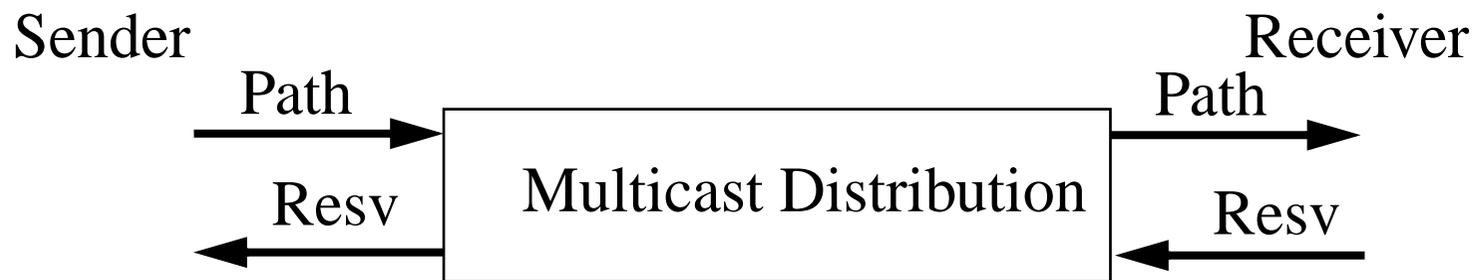
RSVP Soft State

- "soft state" in hosts and routers
- create by PATH and RESV messages
- refreshed by PATH and RESV messages
- Time-outs clean up reservations
- Removed by explicit "tear-down" messages

Maguire
maguire@it.kth.se

RSVP Soft State
2006.03.13

Multicasting and RSVP 483 of 493
Internetworking/Internetteknik

RSVP operation

Figure 88:

Figure 89:

Maguire
maguire@it.kth.se

RSVP operation
2006.03.13

Multicasting and RSVP 484 of 493
Internetworking/Internetteknik
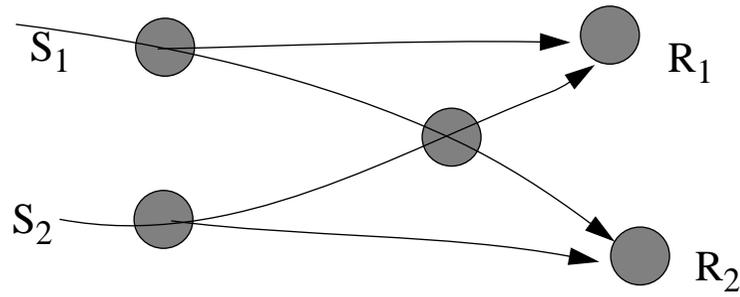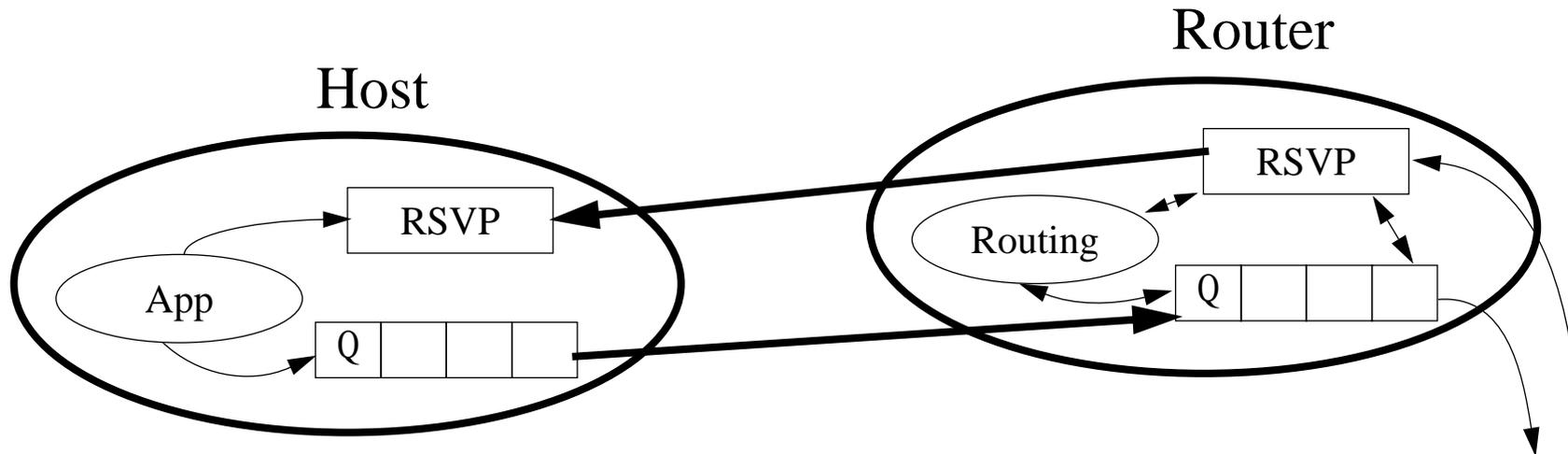
RSVP operations (continued)

- At each node, RSVP applies a local decision procedure "admission control" to the QoS request. If the admission control succeeds, it set the parameters to the classifies and the packet schedule to obtain the desired QoS. If admission control fails at any node, RSVP returns an error indication to the application.

- Each router in the path capable of resource reservation will pass incoming data packets to a packet classifier and then queue these packet in the packet scheduler. The packet classifier determines the route and the QoS class for each packet. The schedule allocates a particular outgoing link for packet transmission.

- The packet schedule is responsible for negotiation with the link layer to obtain the QoS requested by RSVP. The scheduler may also negotiate a "CPU time".

Maguire
maguire@it.kth.se

RSVP operations (continued)
2006.03.13

Multicasting and RSVP 485 of 493
Internetworking/Internetteknik

RSVP Summary

- RSVP supports multicast and unicast data delivery
- RSVP adapts to changing group membership and routes
- RSVP reserves resources for simplex data streams
- RSVP is receiver oriented, i.e., the receiver is responsible for the initiation and maintenance of a flow
- RSVP maintains a "soft-state" in routers, enabling them to support gracefully dynamic memberships and automatically adapt to routing changes
- RSVP provides several reservation models
- RSVP is transparent for routers that do not provide it

Maguire
maguire@it.kth.se
RSVP Summary
2006.03.13
Multicasting and RSVP 486 of 493
Internetworking/Internetteknik

Argument against Reservation

Given, the US has 126 million phones:

- Each conversation uses 64 kbit/sec per phone

- Therefore the total demand is: 8×10^{12} b/s (1 Tbyte/s)

One optical fiber has a bandwidth of ~25×10^{12} b /s

There are well over 1000 transcontinental fibers!

Why should bandwidth be a problem?

Maguire
maguire@it.kth.se

Argument against Reservation
2006.03.13

Multicasting and RSVP 487 of 493
Internetworking/Internetteknik

Further reading

IETF *Routing Area,* especially:

- Inter-Domain Multicast Routing (*idmr*)
- Multicast Extensions to OSPF (*mospf*)

IETF *Transport Area* especially:

- Differentiated Services (*diffserv*)
- RSVP Admission Policy (*rap*)
- Multicast-Address Allocation (*malloc*)

With lots of traditional broadcasters and others discovering multicast -- it is going to be an exciting area for the next few years.

Maguire
maguire@it.kth.se

Further reading
2006.03.13

Multicasting and RSVP 488 of 493
Internetworking/Internetteknik

Summary

This lecture we have discussed:

- Multicast, IGMP, RSVP

Maguire
maguire@it.kth.se

Summary
2006.03.13

Multicasting and RSVP 489 of 493
Internetworking/lnternetteknik

References

[60] Joe Abley, f.root-servers.net, NZNOG 2005, February 2005, Hamilton, NZ

http://www.isc.org/pubs/pres/NZNOG/2005/F%20Root%20Server.pdf

[61] S. Deering, "Host Extensions for IP Multicasting", IETF RFC 1112, August 1989 *http://www.ietf.org/rfc/rfc1112.txt*

[62] W. Fenner, "Internet Group Management Protocol, Version 2", IETF RFC 2236 , November 1997 *http://www.ietf.org/rfc/rfc2236.txt*

[63] B. Cain, S. Deering, I. Kouvelas, B. Fenner, and A. Thyagarajan, "Internet Group Management Protocol, Version 3", IETF RFC 3376, October 2002

http://www.ietf.org/rfc/rfc3376.txt

[64] J. Moy, "Multicast Extensions to OSPF", IETF RFC 1584, March 1994 http://www.ietf.org/rfc/rfc1584.txt

[65] D. Waitzman, C. Partridge, and S. Deering, "Distance Vector Multicast Routing Protocol", IETF RFC 1075 , November 1988

Maguire
maguire@it.kth.se

References
2006.03.13

Multicasting and RSVP 490 of 493
Internetworking/Internetteknik

`http://www.ietf.org/rfc/rfc1075.txt`

[66] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma, and L. Wei, "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification", IETF RFC 2362, June 1998 `http://www.ietf.org/rfc/rfc2362.txt`

[67] A. Adams, J. Nicholas, and W. Siadak, "Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification (Revised)", IETF RFC 3973, January 2005 `http://www.ietf.org/rfc/rfc3973.txt`

[68] D. Meyer and P. Lothberg, "GLOP Addressing in 233/8", IETF RFC 3180 September 2001 `http://www.ietf.org/rfc/rfc3180.txt`

[69] T. Bates, Y. Rekhter, R. Chandra, and D. Katz, "Multiprotocol Extensions for BGP-4", IETF RFC 2858, June 2000 `http://www.ietf.org/rfc/rfc2858.txt`

[70] Beau Williamson, *Developing IP Multicast Networks*, Cisco Press, 2000

[71] Internet Protocol Multicast, Cisco, Wed Feb 20 21:50:09 PST 2002

Maguire
maguire@it.kth.se

References
2006.03.13

Multicasting and RSVP 491 of 493
Internetworking/Internetteknik

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ipmulti.htm

[72] B. Fenner and D. Meyer (Editors), "'Multicast Source Discovery Protocol (MSDP)", IETF RFC 3618, October 2003 *http://www.ietf.org/rfc/rfc3618.txt*

[73] T. Speakman, J. Crowcroft, J. Gemmell, D. Farinacci, S. Lin, D. Leshchiner, M. Luby, T. Montgomery, L. Rizzo, A. Tweedly, N. Bhaskar, R. Edmonstone, R. Sumanasekera and L. Vicisano, "PGM Reliable Transport Protocol Specification", IETF RFC 3208 , December 2001

[74] S. Bhattacharyya (Ed.), "An Overview of Source-Specific Multicast (SSM)", IETF RFC 3569, July 2003 *http://www.ietf.org/rfc/rfc3569.txt*

[75] D. Meyer, "Administratively Scoped IP Multicast", IETF RFC 2365, July 1998 *http://www.ietf.org/rfc/rfc2365.txt*

[76] B. Quinn and K. Almeroth, "IP Multicast Applications: Challenges and Solutions", IETF RFC 3170,September 2001 *http://www.ietf.org/rfc/rfc3170.txt*

[77] R. Braden (Ed.), L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource

Maguire
maguire@it.kth.se

References
2006.03.13

Multicasting and RSVP 492 of 493
Internetworking/Internetteknik

ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", IETF RFC 2205, September 1997 *http://www.ietf.org/rfc/rfc2205.txt*

[78] Y. Snir, Y. Ramberg, J. Strassner, R. Cohen, and B. Moore, "Policy Quality of Service (QoS) Information Model", IETF RFC 3644, November 2003

http://www.ietf.org/rfc/rfc3644.txt

Maguire
maguire@it.kth.se

References
2006.03.13

Multicasting and RSVP 493 of 493
Internetworking/Internetteknik

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 9: Applications: Network Management and VoIP

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapters

KTH Information and Communication Technology

Lecture 5: Outline

- Network Management
- SNMP
- VoIP

Maguire
maguire@it.kth.se
Lecture 5: Outline   Applications: Network Management and VoIP 495 of
2006.03.13
Internetworking/Internetteknik

ISO FCAPS Network Management Model

- Fault management

- Configuration management

- Accounting management

- Performance management

- Security management

Maguire
maguire@it.kth.se

ISO FCAPS Network Management Model
2006.03.13

Applications: Network Management and
Internetworking/Internetteknik

Network Management Process

Network Management System (NMS)

Decision

Management entity

Reporting

Control

network

Agent

Agent

Agent

Information Collecting

Management DB

Management DB

Management DB

proxy

device

device

device

Managed devices

Maguire
maguire@it.kth.se

Network Management Process
2006.03.13

Applications: Network Management and VoIP
Internetworking/Internetteknik
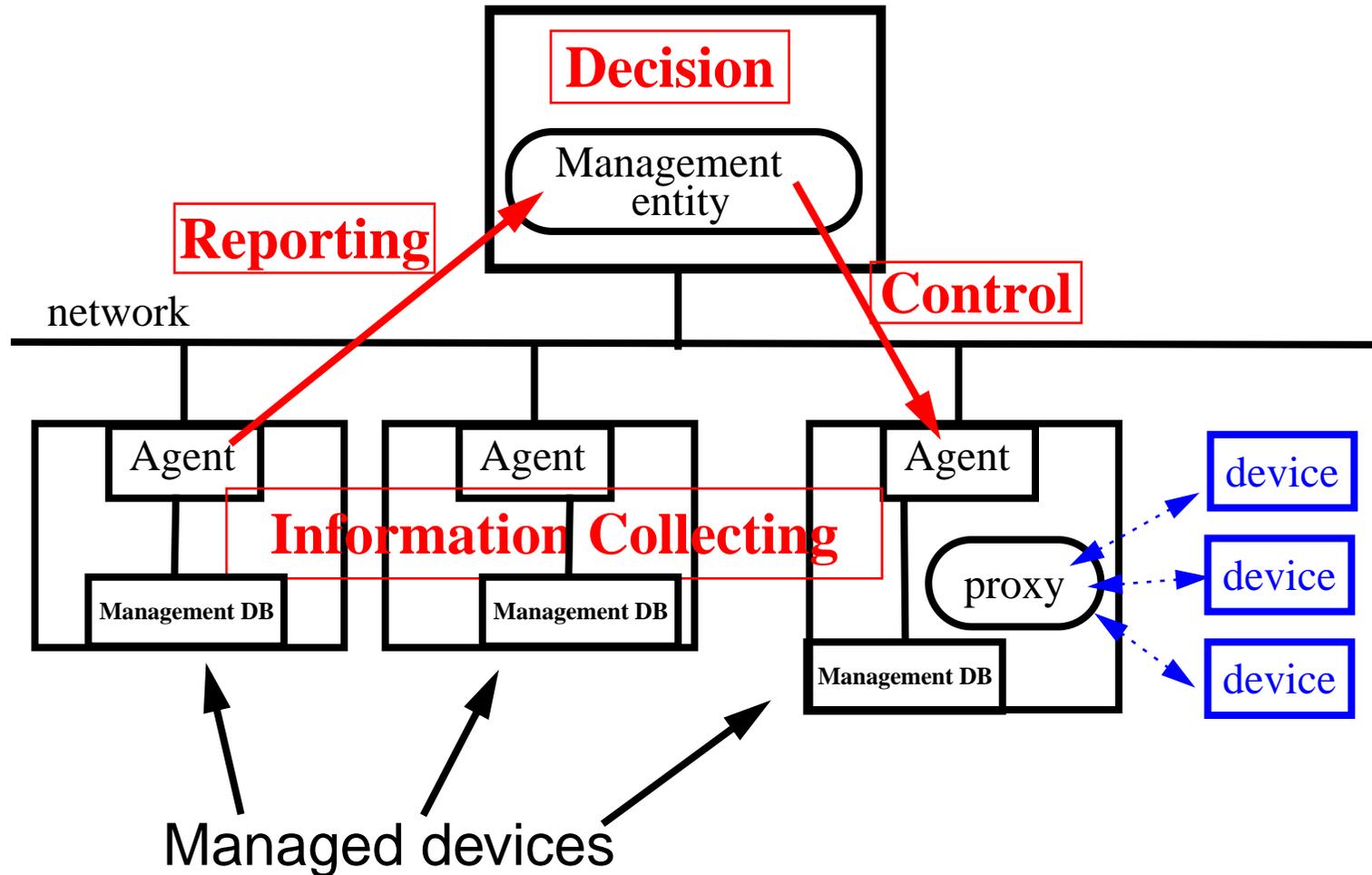
Network Management Process

Simple Network Management Protocol (SNMP), Management Information Base (MIB), Sturecture of Management Information (SMI)

Network Management System (NMS)

Maguire
maguire@it.kth.se

Network Management Process
2006.03.13

Applications: Network Management and VoIP
Internetworking/Internetteknik
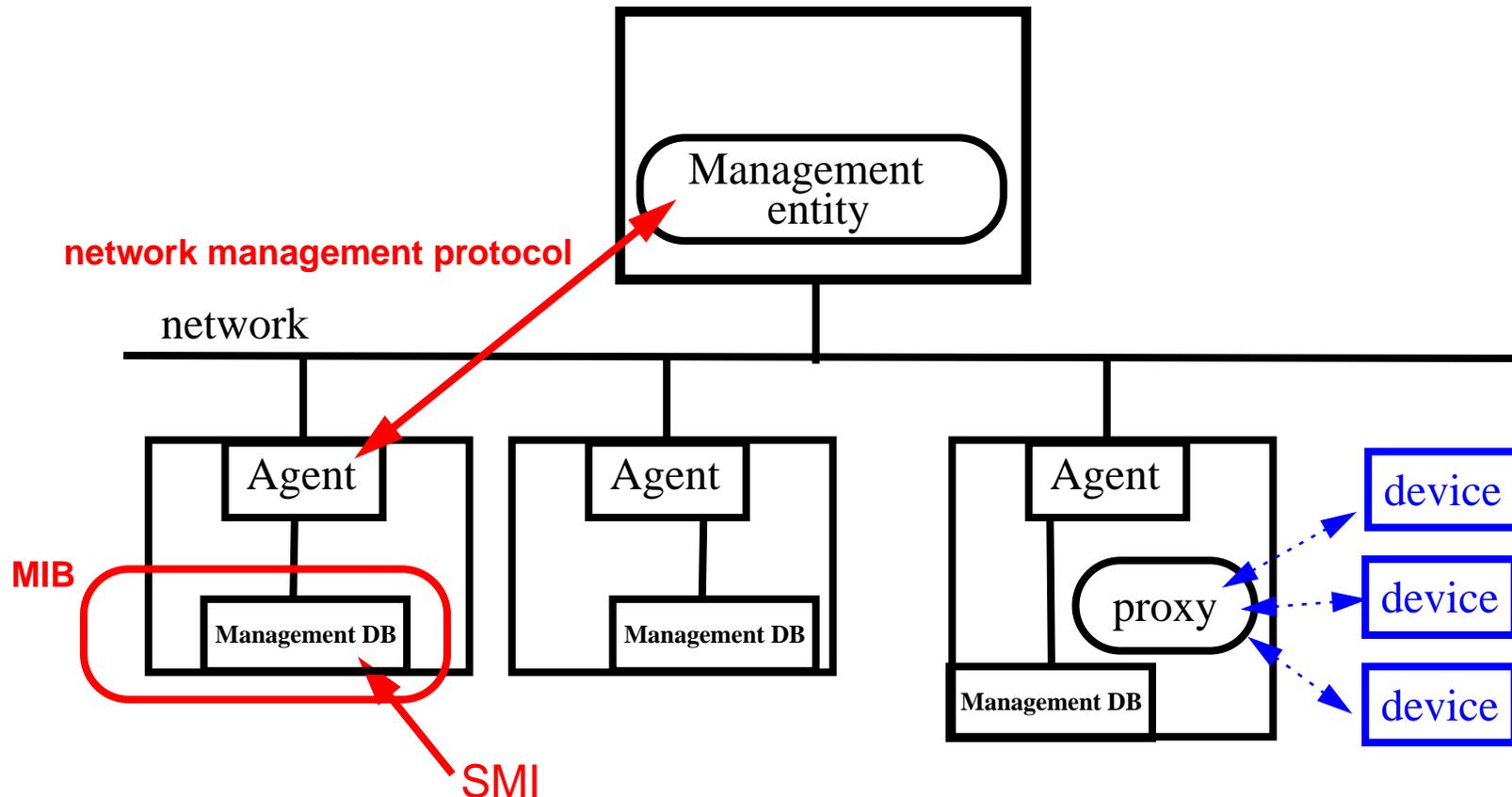
SNMP

Version 1

Version 2 - in 1992-1993, the SNMPv2 Working Group developed a security model based on parties to an SNMP transaction - this was known as SNMPv2p. But the working group decided that a user-based security model was much simpler - and hence more likely to be deployed.

December 1995, the SNMPv2 Working Group was deactivated, but two prominent approaches emerged from independent groups:

SNMPv2u	early standardization of the security features and a minimal specification - to encourage rapid deployment of simple agents;
	deferred standardization of features for managing large networks
SNMPv2*	concurrent standardization of **security** and **scalability** features to ensure that the security design addressed issues of: proxy, trap destinations, discovery, and remote configuration of security
	Focus was effective management of medium and large networks.

August 1996 a team was formed to recommend a single approach.

Maguire
maguire@it.kth.se

SNMP
2006.03.13

Applications: Network Management and VoIP 499 of 561
Internetworking/Internetteknik

SNMPv3

March 1997, the `SNMPv3 Working group` was chartered to define a standard for SNMP security and administration. Target: April 1998 - all SNMPv3 specifications submitted to IESG for consideration as Proposed Standards.

Based on "An Architecture for Describing SNMP Management Frameworks" (`RFC 2271`)

Composed of multiple subsystems:

1. a message processing and control subsystem - Message Processing and Dispatching for SNMP (`RFC 2272`)

2. a security subsystem - based on a User-based Security Model (USM) (`RFC 2274`), provides SNMP message level security (Keyed-MD5 as the authentication protocol and the use of CBC-DES as the privacy protocol - but with support for others) defines a MIB for remotely monitoring/managing the configuration parameters for this Security model

3. a local processing subsystem - responsibile for processing the SNMP PDUs that operate on local instrumentation, applies access control [View-based Access Control Model (VACM) (`RFC 2275`)] and invokes method routines to access management information, and prepares a response to the received SNMP request.

4. SNMPv3 Applications (`RFC 2273`) - includes Proxy Forwarder Applications, which can forward SNMP requests to other SNMP entities, to translate SNMP requests of one version into SNMP requests of another version or into operations of some non-SNMP management protocol; and support aggregated managed objects where the value of one managed object depends upon the values of multiple (remote) items.

Maguire
maguire@it.kth.se
SNMPv3   Applications: Network Management and VoIP 500 of 561
2006.03.13
Internetworking/Internetteknik

SNMP

- ## SNMPv1
  - only 5 commands: get-request, get-next request, set-request, response
  - Clear-text password

- ## SNMPv2: 1992-1996
  - get-bulk-request
  - inform-request (for proxy)
  - trap
  - v2 MIB and M2M MIB
  - Authentication

- ## SNMPv3: 1997-
  - more security enhancement
  - View-based access control - so different managers can see different subset of the information
  - remote configuration

Maguire
maguire@it.kth.se

SNMP
2006.03.13

Applications: Network Management and VoIP 501 of 561
Internetworking/Internetteknik

Management Information Base: MIB

MIB is the database of information maintained by the agent that the manager can query or set.

It specifies the data items a managed device must keep, the operations allowed on each item.

See RFC 1213 "Management Information Base for Network Management of TCP/IP-based internets: MIB-II" *http://www.ietf.org/rfc/rfc1213.txt*

See also:

- RFC 2011: SNMPv2 Management Information Base for the Internet Protocol using SMIv2.
- RFC 2012: SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2.
- RFC 2013: SNMPv2 Management Information Base for the User Datagram Protocol using SMIv2.

Case Diagram

To understand the relationship between counters and to make sure that all the data paths for a packet are accounted for.

Application Layer

udpInDatagrams

udpNoPorts

udpOutDatagrams

udpInErrors

IP Layer
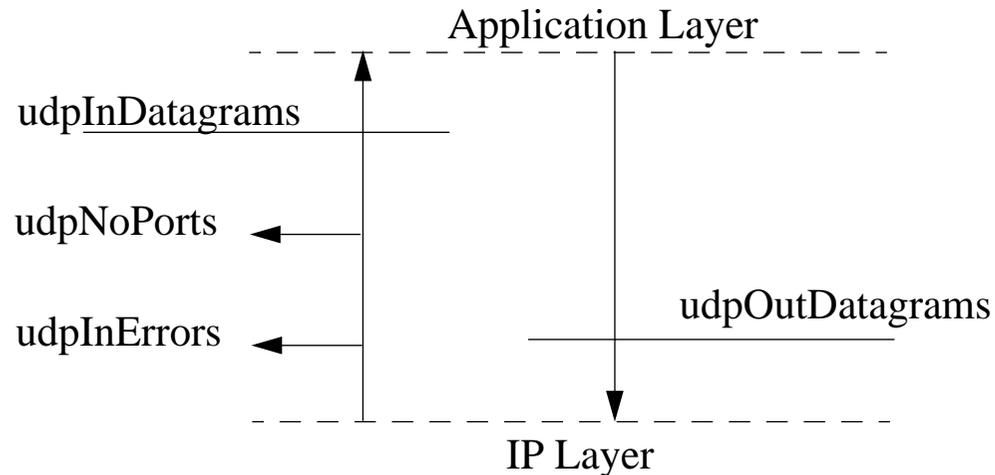
Figure 90: Case diagram of UDP group (W. R. Stevens, TCP/IP Illustrated, V.1, pg. 367)

SNMP Traps

Agent sends a trap to manager to indicate that something has happened.

Following trap types are defined:

0: coldStart

1: warmStart

2: linkDown

3: linkUP

4: authenticationFailure

5: egpNeighborLoss

6: enterpriseSpecific

SNMP agent

bsdi   sun

trap

.35   .33
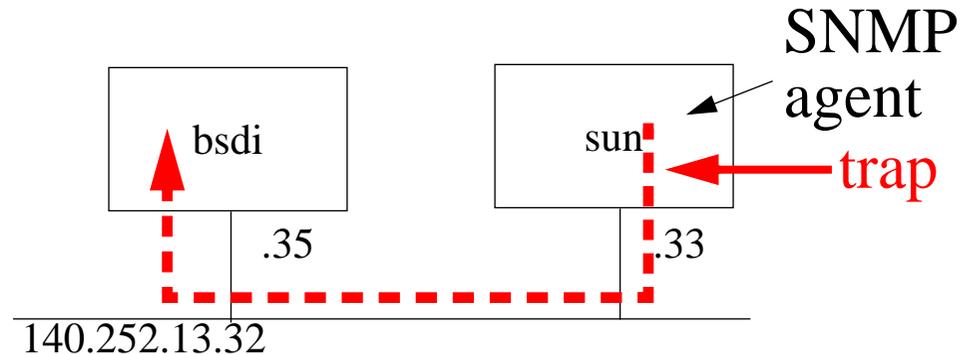
140.252.13.32

Example: start the SNMP agent on sun and send traps to bsdi; tcpdump output:

Port 162   Port 161

trap type   PDU type (length)

1 0.0

sun.snmp > bsdi.snmp-trap: C=traps Trap(28)

E:unix.1.2.5 [140.252.13.33] coldstart 20   timestamp

2 18.86   (18.86)

sun.snmp > bsdi.snmp-trap: C=traps Trap(29)

E:unix.1.2.5 [140.252.13.33] authenticationFailure 1907

Enterprise: sysObjectID   IP address of agent

Remote MONitoring (RMON)

[*RMON MIB 1 (RFC1757)*, *RMON MIB 2 (RFC 2021)*, *RMON MIB Protocol Identifiers (RFC 2074)*, *MIB II (RFC1213)*]

Standard way for users to proactively manage multiple LANs from a central site.

RMON 1

- Notify manager of errors
- provide alerts for network problems
- collects statistical baseline data (i.e., what is "normal" on this LAN), and
- acts as a remote network analyser.

RMON 2

- access higher level protocol information,
- Point-to-point traffic statistics broken down by higher layer protocols,
- eases trouble-shooting, and
- enables network capacity planning [and to solve problems before they become problems].

RMON Probes or Monitors

Network monitoring devices (monitor or probes) are instruments that exist for the purpose of managing a network. Essentially a LAN analyzer - which is always connected to the segment.

- A physical device which is attached to a segment of the network (it will promisciously listen to traffic - to collect statistics and if requested packets)

- Generally a microprocessor based system with 8^+MBytes of memory.

- Fairly powerful processors so that events and alarms are not missed.

- In-band or out-of-band communication
  - In-band - you communicate via the probe via the segment it is monitoring
  - Out-of-band - you communicate with it via another path, e.g., a PPP/SLIP/serial connection

- Probes can operate off-line, i.e., they operate even though they may not be in contact with the network managment system.

Probes are sold by lots of vendors.

Maguire
maguire@it.kth.se
RMON Probes or Monitors Applications: Network Management and VoIP 506
2006.03.13
Internetworking/Internetteknik

RMON1 Statistics

Information collected by examing MAC layer

		Group	Description
Tables	1	Statistics	Statistics for the segment to which the RMON probe is attached
	2	History	History (Baselines) of the segment
	4	Host	Per host statistics for each individual transmitting and receiving device.
	5	Host Top N	Top N reports on base statistics.
	6	Matrix	Statistics on all conversations (i.e., who talks to whom)
Packet Capture	7	Filter	Match on any part of a frame, including errors (CRC, overruns, etc.)
	8	Capture	Collect packets, based on filters, for later retrieval (as if you were a network analyzer
SNMP traps	3	Alarm	Alarms to monitor for user-defined events.
	9	Event	Log file for use in conjunction with the Alarm or Filter Group.
Token rings		Token Ring	Ring Station Order, Ring Configuration and Source Routing Information.

Defined by *RFC 1757.*

Ethernet Statistics Group

"These statistics take the form of free running counters that start from zero when a valid entry is created. Each etherStatsEntry contains statistics for one Ethernet interface. The probe must create one etherStats entry for each monitored Ethernet interface on the device." - from RFC1757

etherStatsTable OBJECT-TYPE

SYNTAX SEQUENCE OF EtherStatsEntry ...   ::= { statistics 1 }

etherStatsEntry OBJECT-TYPE

DESCRIPTION

"A collection of statistics kept for a particular Ethernet interface. As an example, an instance of the etherStatsPkts object might be named etherStatsPkts.1"

INDEX { etherStatsIndex }   ::= { etherStatsTable 1 }

EtherStatsEntry

EtherStatsEntry ::= SEQUENCE {

    etherStatsIndex   INTEGER (1..65535),

etherStatsDataSource   OBJECT IDENTIFIER,
etherStatsDropEvents   Counter,
etherStatsOctets   Counter,
etherStatsPkts   Counter,
etherStatsBroadcastPkts   Counter,
etherStatsMulticastPkts   Counter,
etherStatsCRCAlignErrors   Counter,
etherStatsUndersizePkts   Counter,
etherStatsOversizePkts   Counter,
etherStatsFragments   Counter,
etherStatsJabbers   Counter,
etherStatsCollisions   Counter,
etherStatsPkts64Octets   Counter,
etherStatsPkts65to127Octets   Counter,
etherStatsPkts128to255Octets Counter,
etherStatsPkts256to511Octets Counter,
etherStatsPkts512to1023Octets Counter,
etherStatsPkts1024to1518Octets Counter,
etherStatsOwner   OwnerString,
etherStatsStatus   EntryStatus

  }

EtherHistoryEntry

EtherHistoryEntry::= SEQUENCE {

etherHistoryIndex	INTEGER (1..65535),
etherHistorySampleIndex	INTEGER (1..2147483647),
etherHistoryIntervalStart	TimeTicks,
etherHistoryDropEvents	Counter,
etherHistoryOctets	Counter,
etherHistoryPkts	Counter,
etherHistoryBroadcastPkts	Counter,
etherHistoryMulticastPkts	Counter,
etherHistoryCRCAlignErrors	Counter,
etherHistoryUndersizePkts	Counter,
etherHistoryOversizePkts	Counter,
etherHistoryFragments	Counter,
etherHistoryJabbers	Counter,
etherHistoryCollisions	Counter,
etherHistoryUtilization	INTEGER (0..10000)

}

HostEntry

HostEntry ::= SEQUENCE {

      hostAddress   OCTET STRING,

    hostCreationOrder   INTEGER (1..65535),

    hostIndex   INTEGER (1..65535),

    hostInPkts   Counter,

    hostOutPkts   Counter,

    hostInOctets   Counter,

    hostOutOctets   Counter,

    hostOutErrors   Counter,

    hostOutBroadcastPkts   Counter,

    hostOutMulticastPkts   Counter

    }

Host Top N group

Used to prepare reports that describe the hosts that top a list **ordered** by one of their statistics.

hostTopNControlTable is used to initiate the generation of such a report, the management station selects the parameters, such as:

- which interface,
- which statistic,
- how many hosts, and
- the start and stop times of the sampling.

Maguire
maguire@it.kth.se

Host Top N group
2006.03.13

Applications: Network Management and VoIP 512 of
Internetworking/Internetteknik

The Matrix Group

Matrix group consists of the matrixControlTable, matrixSDTable and the matrixDSTable.

These tables store statistics for a particular conversation between two addresses.

The maxtrixSDTable - contains a entries indexed by source and destination.

MatrixSDEntry ::= SEQUENCE {

       matrixSDSourceAddress   OCTET STRING,

       matrixSDDestAddress   OCTET STRING,

       matrixSDIndex   INTEGER (1..65535),

       matrixSDPkts   Counter,

       matrixSDOctets   Counter,

       matrixSDErrors   Counter

       }

The maxtrixSDTable - a similar set of statistics (MatrixDSEntry) indexed by destination and source.

Maguire
maguire@it.kth.se

The Matrix Group
2006.03.13

Applications: Network Management and VoIP 513 of
Internetworking/Internetteknik

RMON2

Information collected from network and higher layer ("application") headers (defined by *RFC2021*)

		Group	Description
Protocols	11	Protocol Directory	List of protocol types the probe is capable of monitoring
	12	Protocol Distribution	Number of packets and octets by protocols on a network segment
Network layer	13	Address Mapping	MAC addresses and corresponding network addresses
	14	Network Layer Host	Amount of traffic sent to and from each network address
	15	Network Layer Matrix	Amount of traffic between each pair of network addresses
		Network Layer Matrix Top N	Top N conversations over a user-defined period (packet or octet counts)
Higher layers	16	Application Layer Host	Amount of traffic, by protocol
	17	Application Layer Matrix	Amount of traffic, by Protocol, between each pair of network addresses.
		Application Layer Matrix Top N	Top N conversations over a user-defined period (packet or octet counts)

Maguire
maguire@it.kth.se

RMON2
2006.03.13

Applications: Network Management and VoIP 514 of 561
Internetworking/Internetteknik

Information collected from network and higher layer ("application") headers (defined by _RFC2021_)

	Group		Description
	18	User History	Users created custom History Tables based on supported OID's.
Probe itself	19	Probe Configuration	Configuration of various operating parameters of the probe
	20	RMON Conformance	Lists which groups and instances of a group a probe supports

Maguire
maguire@it.kth.se

RMON2
2006.03.13

Applications: Network Management and VoIP 515 of 561
Internetworking/Internetteknik

Proprietary MIBs to extend RMON functions

ION Network, Inc. adds:

Group	Description
FDDI	FDDI MAC level and User Data Statistics for FDDI networks
Protocol	Bandwidth utilization by protocols
SolCom Host	Tracks MAC to IP address mappings; including when a host was first and last seen, when a new host appears on the segment
Traffic Generation	Generate traffic using user-defined packets (including packet with errors)
Response Time Monitoring	Works out response times and helps to pin-point WAN failures using ICMP echo-requests initiated from the central site.

Maguire
maguire@it.kth.se

Proprietary MIBs to extend RMON functions   Applications: Network Management and
2006.03.13   Internetworking/Internetteknik

Network Management Systems

- HP OpenView -- *http://www.managementsoftware.hp.com/marketsegments/enterprises/index.asp*
  - Derived from OpenView: IBM NetView, Digital Polycenter NetView, and NCR OneVision
- SunSoft Solstice: Site Manager, Solstice Domain Manager, and Enterprise Manager -- *http://www.sun.com/solstice/index.html*
- Aprisma Management Technologies' Spectrum *http://www.aprisma.com/*

Maguire
maguire@it.kth.se

Network Management Systems
2006.03.13

Applications: Network Management and VoIP
Internetworking/Internetteknik

WEB based Management

Using the Web as an interface

- ## Web based Reporting/Statistics
  - Netscount *http://www.netscout.com/*
  - HP*Netmetrix* WebReporter
  - Network Statistics Collection And Reporting Faclility (Netscarf
    http://www.merit.edu/internet/net-research/netscarf/index.html), their Scion package consists of
    five components:
    - scollect - collects network data from a set of routers
    - scook - preprocesses network data into a more convenient (condensed) form
    - scserver - delivers the network data in response to client requests
    - sclient - requests network data from the scserver on behalf of a reporting or graphing
      application
    - Real-Time Data (rtdata) tree - a flat-file database: stores the data collected by scollect
  - Merit Internet Performance and Analysis Project (*IPMA*), tools: NetNow, AS Explorer, Route
    Flap, Routing Table Statistics Generator, …
    - See also pointers to tools developed by *others*.

- ## Web based Interfaced Management Platforms
  - *OpenView World Wide Web Interface*
  - *DR-Web Manager* and Agent
  - *SiteScope* v2.2 - a Java-based Web Site Monitoring and Administration Software

- ## Web based Interfaced Management Tools
  - Cisco *ClickStart* - for configuring a Router with a Web Browser
  - Axis Communications AB's *Thin Server*

- ## Management of Web Services
  - Harrie Hazewinkel, Carl W. Kalbfleisch, Juergen Schoenwaelder, "Definitions of Managed Objects for WWW Services", November 11, 1997 *http://ietf.org/ids.by.wg/applmib.html*
    - Service Information Group
    - Protocol Statistics Group
    - Document Statistics Group

Web Based Enterprise Management Initiative (WBEM)

see `http://www.dmtf.org/wbem`

Goal: to consolidate and unify the data provided by **existing** management technologies - in order to solve enterprise problems; i.e., from the application layer problem report down to the interface card - even if the card is in a remote branch office.

Builds on:Intel's Wired for Management (WfM) effort ==> `Distributed Management Task Force` (formerly Desktop Management Task Force) and Desktop Management Interface (now DMI 2.0)

The DMI was designed to be:

- "independent of a specific computer or operating system
- independent of a specific management protocol
- easy for vendors to adopt
- usable locally -- no network required
- usable remotely using DCE/RPC, ONC/RPC, or TI/RPC
- mappable to existing management protocols (e.g., CMIP, SNMP)
- The DMI procedural interfaces are specifically designed to be remotely accessible through the use of Remote Procedure Calls. The RPCs supported by the DMI include: DCE/RPC, ONC/RPC, and TI/RPC."   -- DMI 2.0 Introduction

DMI 2.0 has three groups:

- ComponentID group - required for all DMI components, includes information such as the six named attributes: "Manufacturer", "Product", "Version", "Serial Number", "Installation", and "Verify" [asking for this last group causes the device to check itself].
- Event Groups
  - includes a template group used to describe the format of event data for standard events
  - Event State group is defined to hold the current state of state-based events
  - Events can be of different severity levels: Monitor, Information, OK, Non-Critical, Critical, and Non-Recoverable.
- DMI Service Provider Groups - provides the means for those interested in specific events to subscript to just the events that they want; subscribers can say how they want to be notified (DCE RPC, TI RPC, ONC RPC), what transport protocol should be used (TCP/IP, IPX, …), when the no longer want to be notified (Subscription Expiration DateStamp), …

Four Elements of DMI

- a format for describing management information - Management Information Format (MIF)
    - a language for describing each component;
    - each component has a MIF file to describe its manageable characteristics; and
    - When a component is initially installed into the system, the MIF is added to the (implementation-dependent) MIF database.

- a service provider entity

- two sets of APIs, one set for service providers and management applications to interact (Service Provider API for Components), and the other for service providers and components to interact (Component Provider API), and

- set of services for facilitating remote communication.

Common Information Model (CIM)

- DMTF Common Information Model (CIM)

  http://www.dmtf.org/standards/standard_wbem.php based on object-oriented technologies for use in Web-based management

- XML Mapping Specification v2.0.0

- XML Document Type Definition v2.0.0

- CIM Operations over HTTP, V1.0

Java and Management

Java Management API (JMAPI)- Set of extensible objects and methods, defines an application programming interfaces (API) which includes:

- JavaManagement API User Interface Style Guide
- Admin View Module (AVM)
- Base Object Interfaces
- Managed Container Interfaces
- Managed Notification Interfaces
- Managed Data Interfaces
- Managed Protocol Interfaces
- SNMP Interfaces
- Applet Integration Interfaces

Java Dynamic Management Kit - A Java agent toolkit for rapid development of autonomous Java agents for system, application, or network devices.

Inter-domain Management task force (XoJIDM)

Sponsored by X/Open and the Network Management Forum (NMF), see

Inter-Domain Management, Open Group Technical Standard, C802 ISBN 1-85912-256-6 January 2000 524 pages.

They have specified such things as SNMP MIBS to CORBA-IDL conversion, CORBA-IDL to GDMO/ASN.1 conversion, CORBA/SNMP Gateway, … .

Maguire
maguire@it.kth.se

Inter-domain Management task force (XoJIDM) Applications: Network Management and
2006.03.13
Internetworking/Internetteknik

Policy Based Management

See the recent exjobb report "Implementing policy-based network management" by Yavor Adel Al-Shaikhly

Check Point's `http://www.checkpoint.com/products/management/index.html` policy management framework

"`Policy Agents: Licensed to Manage/Policy Based Management of Distributed Systems`" by Morris Sloman. Department of Computing, Imperial College, London, U.K.

Cisco's "`CiscoAssure Policy Networking: Enabling Business Applications through Intelligent Networking`"

Applications

- E-mail
  - E-mail was invented by Ray Tomlinson of BBN in 1972.
  - His e-mail utility program permits listing, forwarding, and responding to e-mails
  - It was demonstrated at International Computer Communication Conference (ICCC) that year.
  - It become the first "killer application" of the Internet.

- Telnet and FTP

- Networked File systems (such as NFS)

- X windowing system

- Web browsers
  - The first graphical Web browser (called Mosaic) is introduced in 1993
  - It was developed at the National Center for Supercomputing at the University of Illinois.

Maguire
maguire@it.kth.se
Applications Applications: Network Management and VoIP 527 of 561
2006.03.13
Internetworking/Internetteknik

Voice over IP (VoIP)

First we will set the context and then we will examine the technical details.

VoIP End-to-End Architecture

IP end-to-end

VoIP server

call/sessionrouting

Cellular IP terminal

Trancoding

Fixed IP terminal

codec

IP

radio

Multiparty
conference Units

codec

IP

ethernet

IP cloud
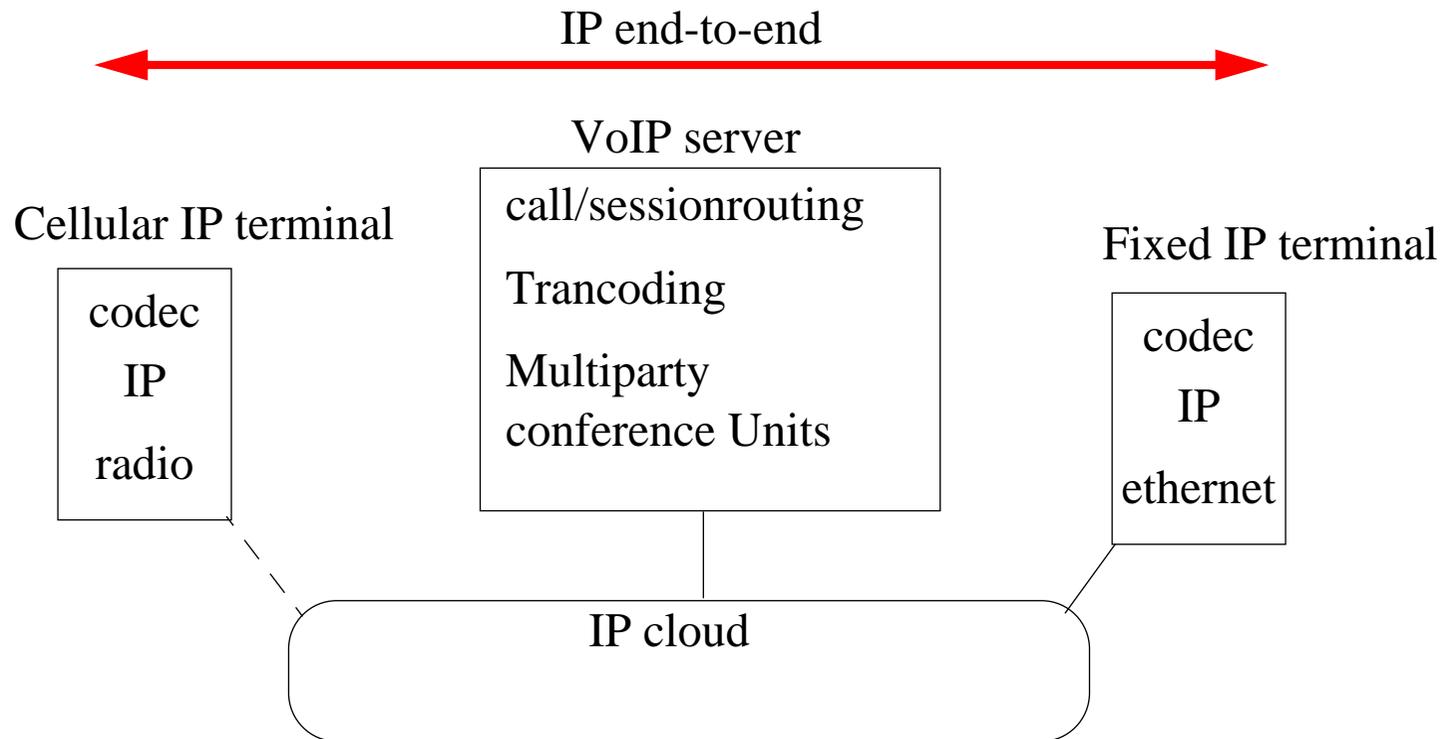
Deregulation $\Rightarrow$ New regulations

- US Telecommunications Act of 1996[1]
  - "The goal of this new law is to let anyone enter any communications business -- to let any communications business compete in any market against any other."[2]
  - updated the Communications Act of 1934

- New interconnection points
  - perhaps there is something that LECs can do with all the empty space in their central exchanges [which appeared due to the shrinking size of their own switching equipment]

- Number portability - even local numbers
  - every call results in ~10 DB lookups

- "Universal Service"
  - from a myth to a legal requirement
  - an evolving service level - not a fixed service or service level!
  - special subsidies for schools, health care, libraries, etc.

- February 1997 World Trade Organization (WTO) agreement[3]

---

1. The official citation for the new Act is: Telecommunications Act of 1996, Pub. LA. No. 104-104, 110 Stat. 56 (1996).

2. http://www.fcc.gov:80/telecom.html

3. For informal background see "WTO negotiations on basic Telecommunications" - http://www.wto.org/wto/services/tel.htm

Maguire
maguire@it.kth.se
Deregulation $\Rightarrow$ New regulations   Applications: Network Management and VoIP
2006.03.13
Internetworking/Internetteknik

Deregulation continued

$\Longrightarrow$ New operators

Lots of new actors as operators

$\Longrightarrow$ New Suppliers

Lots of new actors as equipment suppliers

Traditional telecom equipment vendors buying datacom vendors:

Lots of mergers and acquisitions among datacom vendors.

Latency
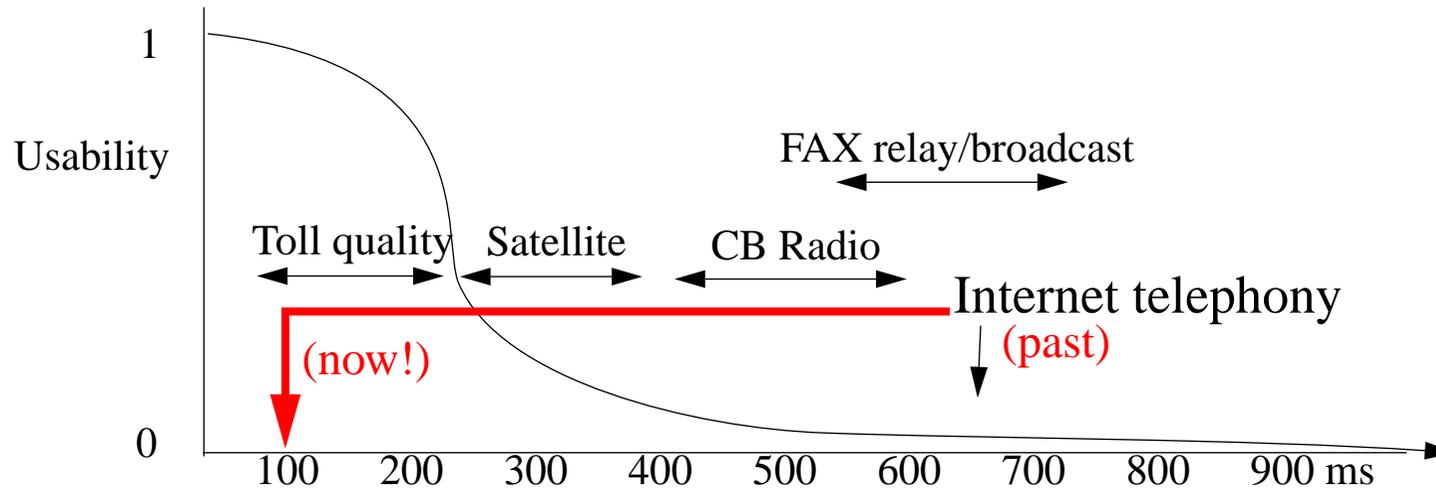
Figure 91: Usability of a voice circuit as a function of end-to-end delay (adapted from a drawing by Cisco)[a]

a. http://www.packeteer.com/solutions/voip/sld006.htm

However:

Round-trip	min (ms)	avg (ms)	max (ms)	hops
Local LAN	1	1	3	0
to northern Sweden (basil.cdt.luth.se)	21	25	41	8
to Austria (freebee.tu-graz.ac.at)	73	109	353	18
To server in US network	131	306	526	19
To my machine in the US (~30 ms is the ISDN link)	175	328	600	21
To KTH's subnet at Stanford University in the US (ssvl.stanford.edu)	166	170	217	20

Increasingly IP based data+voice infrastructure

Public cells

Handset

exchange

E-1 to PSTN

Voice gateway

ISDN-PRI

DECT

KTH/Monaco GSM system

GSM

Handset

Handset

Workstation

Home

Voice Gateway

Workstation

Office

Router

Internet

Basestation

Mobile
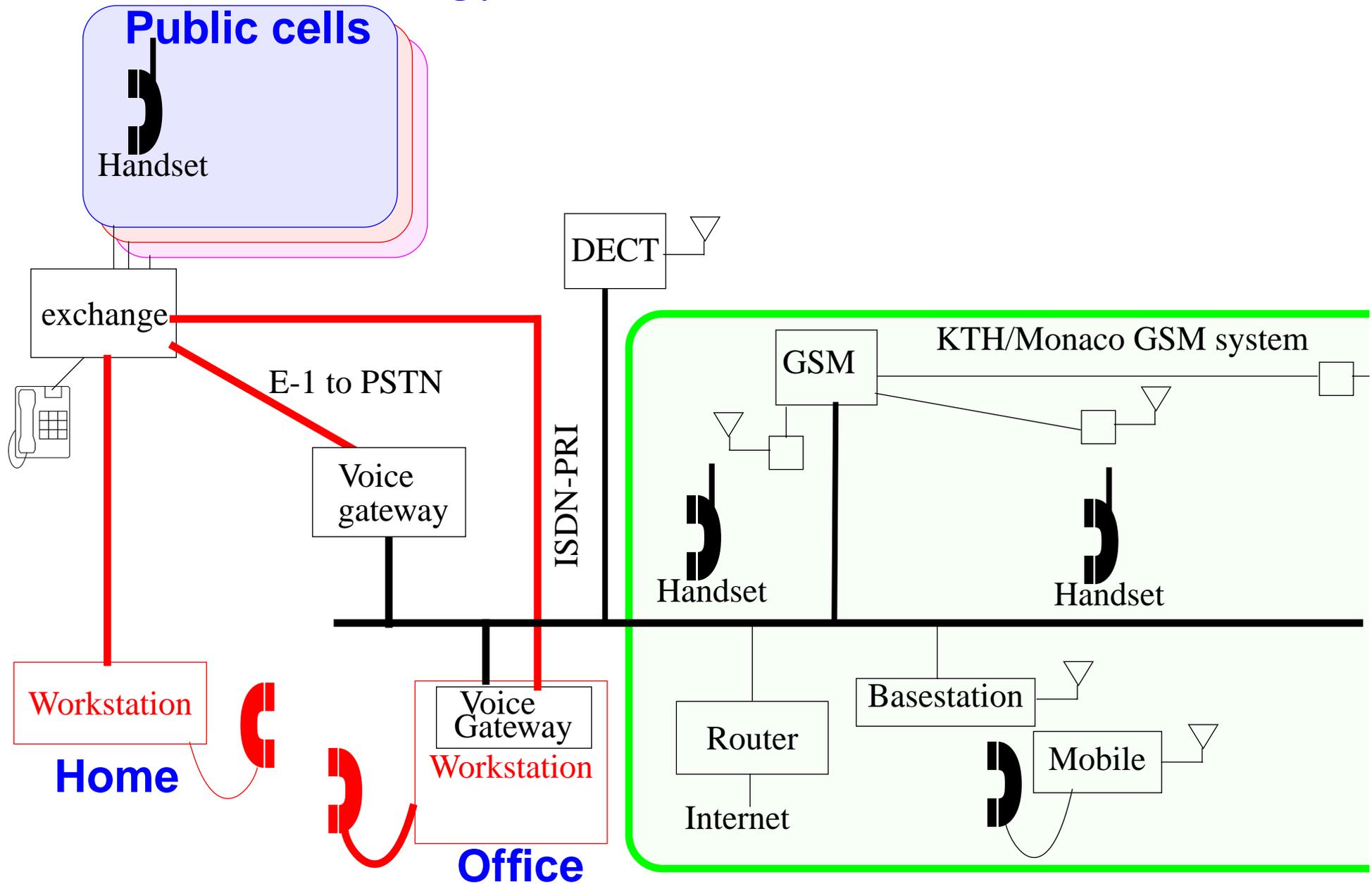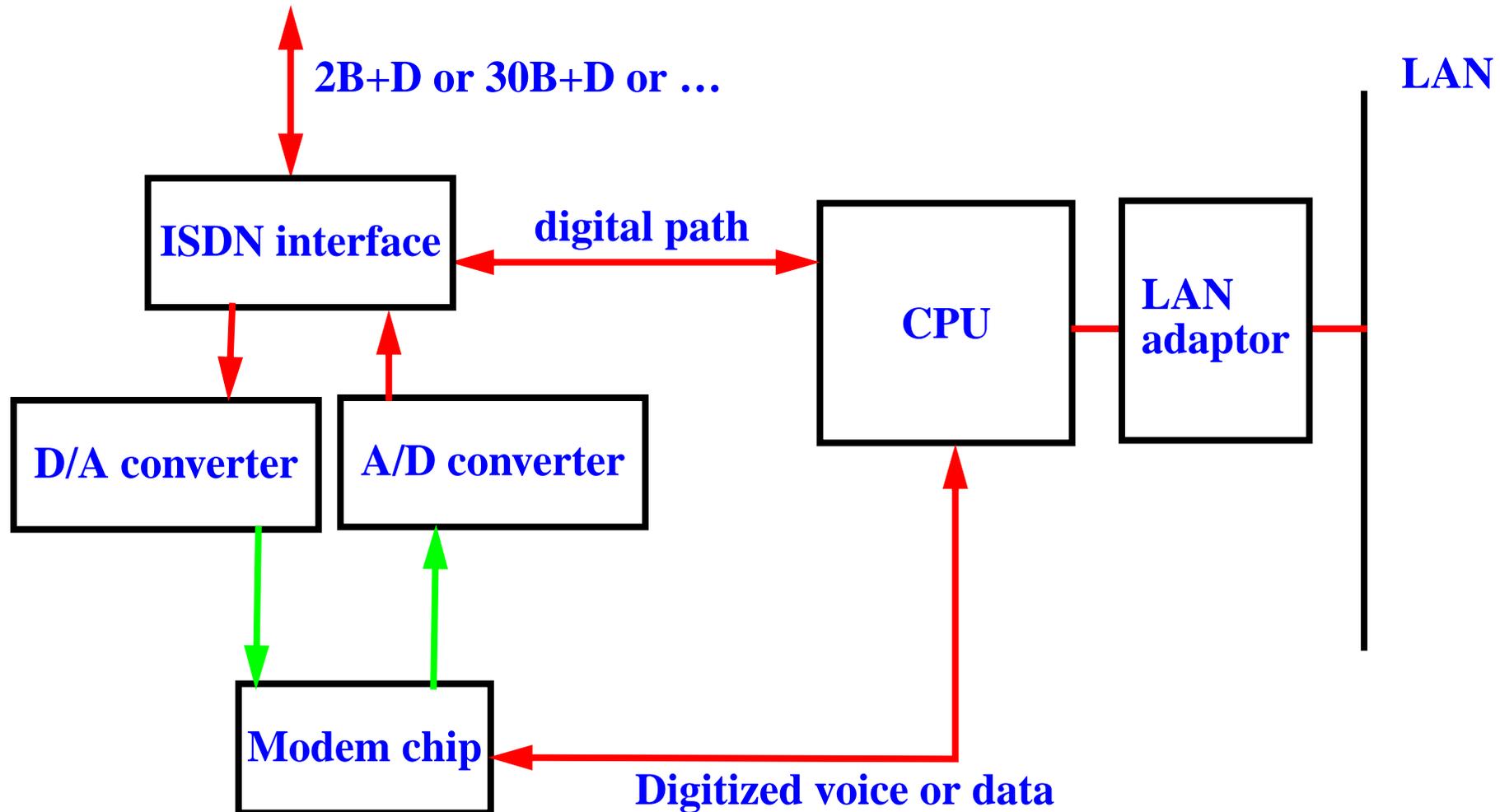
Voice Gateway

Use access servers such as Ascend Communications MAX, filled with digital modems (currently used for current analog modem pools) as voice gateways [see Ascend's MultiVoice Application for the MAX]

Voice over IP (VOIP)

Gateways not only provide basic telephony and fax services but can also will enable lots of value-added services, e.g., call-centers, integrated messaging, least-cost routing, … .

Such gateways provide three basic functions:

- Interface between the PSTN network and the Internet

  Terminate incoming synchronous voice calls, compress the voice, encapsulate it into packets, and send it as IP packets. Incoming IP voice packets are unpacked, decompressed, buffered, and then sent out as synchronous voice to the PSTN connection.

- Global directory mapping

  Translate between the names and IP addresses of the Internet world and the E.164 telephone numbering scheme of the PSTN network.

- Authentication and billing

Voice representation

ITU G.723.1 algorithm for voice encoding/decoding or G.729 (CS-ACELP voice compression).

Signaling

Based on the H.323 standard on the LAN and conventional signaling will be used on telephone networks.

Fax Support

Both store-and-forward and real-time fax modes - with store-and-forward the system records the entire FAX before transmission.

Management

Full SNMP management capabilities via MIBs (Management Information Base) will be provided to control all functions of the Gateway. Extensive statistical data will be collected on dropped calls, lost/resent packets, and network delays.

Compatibility

De jure standards:
- ITU G 723.1/G.729 and H.323
- VoIP Forum IA 1.0

De facto standards:
- Netscape's Cooltalk
- Microsoft's NetMeeting

A protocol to keep you eyes on: Session Initiation Protocol (SIP) [RFC 2543], much simpler than H.323

Maguire
maguire@it.kth.se
Voice over IP (VOIP)   Applications: Network Management and VoIP 535 of
2006.03.13
Internetworking/Internetteknik

VOIP Modes of Operation

- PC to PC
- PC-to-Telephone calls
- Telephone-to-PC calls
- Telephone-to-Telephone calls via the Internet
- Premises to Premises
  - use IP to tunnel from one PBX/Exchange to another
- Premises to Network
  - use IP to tunnel from one PBX/Exchange to a gateway of an operator

Maguire
maguire@it.kth.se
Voice over IP (VOIP)   Applications: Network Management and VoIP 536 of
2006.03.13
Internetworking/Internetteknik

Cisco Voice Over IP

Enables Cisco 3600 series routers to carry live voice traffic (e.g., telephone calls and faxes) over an IP network.

They state that this could be used for:

- "Toll bypass
- Remote PBX presence over WANs
- Unified voice/data trunking
- POTS-Internet telephony gateways"

Uses Real-Time Transport Protocol (RTP) for carrying packetized audio and video traffic over an IP network.

Cisco 3600 supports a selection of CODECs:

- G.711 A-Law 64,000 bits per second (bps)
- G.711 u-Law 64,000 bps
- G.729 8000 bps

Maguire
maguire@it.kth.se
Cisco Voice Over IP   Applications: Network Management and VoIP 537 of
2006.03.13
Internetworking/Internetteknik

Cisco 3800 supports even more CODECs:

- ITU G.726 standard, 32k rate
- ITU G.726 standard, 24k rate
- ITU G.726 standard, 16k rate
- ITU G.728 standard, 16k rate (default)
- ITU G.729 standard, 8k rate

By using Voice Activity Detection (VAD) - you only need to send traffic if there is something to send.

An interesting aspect is that user's worry when they hear absolute silence, so to help make them comfortable it is useful to play noise when there is nothing to output. Cisco provide a "**comfort-noise** command to generate background noise to fill silent gaps during calls if VAD is activated".

Cisco 3600 series router can be used as the voice gateway with software such as Microsoft NetMeeting.

Cisco 3800 also supports "fax-relay" - at various rates either current voice rate or

2,400/4,800/7,200/9,600/14,400 bps bps fax rates.

 For futher information see

`http://www.cisco.com/univercd/cc/td/doc/product/softwa`
`re/ios113ed/113t/113t_1/voip/config.htm`

Intranet Telephone System

On January 19, 1998, `Symbol Technologies` and Cisco Systems announced that they had combined the Symbol Technologies' NetVision™ wireless LAN handset and Cisco 3600 to provide a complete wireless local area network telephone system based on Voice-Over-IP technology. (`White Paper`)

The handset use wireless LAN (IEEE 802.11) infrastructure and a voice gateway via Cisco 3600 voice/ fax modules. The system conforms to H.323.

> "I believe that this is the first wireless local area network telephone based on this technology" -- Jeff Pulver

Seamless roaming via Symbol's pre-emptive roaming algorithm with load balancing.

Claim each cell can accommodate ~25 simultaneous, full-duplex phone calls.

Current Ericsson is a partner with Symbol, using Ericsson's `WebSwitch2000`

Maguire
maguire@it.kth.se

Intranet Telephone System
2006.03.13

Applications: Network Management and VoIP 540
Internetworking/Internetteknik

Wireless LANs

"The wireless workplace will soon be upon us[1]

Telia has strengthened its position within the area of radio-based data solutions through the acquisition of Global Cast Internetworking. The company will primarily enhance Telia Mobile's offering in wireless LANs and develop solutions that will lead to the introduction of the wireless office. A number of different alternatives to fixed data connections are currently under development and, *later wireless IP telephony will also be introduced.*

…

The acquisition means that Telia Mobile has secured the resources it needs to maintain its continued expansion and product development within the field of radio-based LAN solutions. *Radio LANs are particularly suitable for use by small and medium-sized companies as well as by operators of public buildings such as airports and railway stations.*

 Today's radio-LAN technology is based on *inexpensive products that do not require frequency certification*. They are *easy to install* and are often used to replace cabled data networks in, for example, large buildings.

…"

[*emphasis* added by Maguire]

---

1. Telia press annoucement: 1999-01-25

Telia's HomeRun

http://www.homerun.telia.com/

A subscription based service to link you to your corporate network from airports, train stations, ferry terminals, hotels, conference centers, etc.

Look for Telia's HomeRun logo:

Carriers offering VOIP

"Equant, a network services provider, will announce tomorrow that it is introducing voice-over-frame relay service in 40 countries, ...

The company says customers can save 20% to 40% or more by sending voice traffic over its frame relay network. "This is the nearest you're going to get to free voice," says Laurence Huntley, executive VP of marketing for Equant Network Service.

The Equant service uses the Cisco Systems 3810 router, which takes the customer's voice and data traffic and integrates them before putting the traffic on the Equant network. Equant is also working with Cisco to introduce a voice-over-IP service. …
Equant isn't alone in its pursuit to send voice traffic over data networks. Most of the major carriers are testing services that would send voice over data networks. ... ."[1]

AT&T VoIP phone: *http://www.telephones.att.com/new_prod.html*

Deutsche Telekom running a pilot Internet telephony service using networking products from Ascend Communications and VocalTec.

---

1. Mary E. Thyfault, Equant To Roll Out Voice-Over-Frame Relay Service, InformationWeek Daily, 10/21/98.

VOIP vs. traditional telephony

In "*Telcos Hear New Voices*" by Margrit Sessions, Phillips Tarifica Ltd., she predicts that by 2001, Internet telephony could squeeze nearly US$1.2 billion in revenue out of 16 international service providers, while losses due to e-mail (US$463 million) and Internet fax (US$170 million) will be much less.

Expected loss of international call revenue due to: Internet phone, fax, and e-mail, by operator:

Company	Expected Losses (millions of US Dollars)	Loss as a percentage of revenue
AT&T	~350	3.6%
Kokusai Denshin Denwa (KDD) Co. Ltd. (Japan)	~307	10.4%
Deutsche Telekom	~175	4.2%
Telstra Corp. (Australia)	~168	9%
Embratel (Brazil)	~28	11.5%
Bezeq (Israel)	~30	10.7%

Economics

"Can Carriers Make Money On IP Telephony? by Bart Stuck and Michael Weingarten, Business Communication Review, Volume 28, Number 8, August 1998, pp. 39-44 - *http://www.bcr.com/bcrmag/08/98p39.htm*

"What is the reality in the battle over packet-versus-circuit telephony, and what is hype?

Looking at the potential savings by cost element, it is clear that in 1998, access arbitrage is the major economic driver behind VOIP. By 2003, we anticipate that switched-access arbitrage will diminish in importance, as the ESP exemption disappears and/or access rates drop to true underlying cost.

However, we believe that the convergence between voice and data via packetized networks will offset the disappearance of a gap in switched access costs. As a result, VOIP will continue to enjoy a substantial advantage over circuit-switched voice. Indeed, as voice/data convergence occurs, we see standalone circuit-switched voice becoming economically nonviable."

Conferences

VON (Voice on the Net)

Patents

Mixing voice and data in the LAN goes back to at least this patent:

4581735 : Local area network packet protocol for combined voice and data transmission

INVENTORS:

Flamm; Lois E., Chatham Township, Morris County, NJ

Limb; John O., Berkeley Heights, NJ

ASSIGNEES: AT&T Bell Laboratories, Murray Hill, NJ

ISSUED: Apr. 8 , 1986

FILED: May 31, 1983

ABSTRACT: In order to control the transfer of packets of information among a plurality of stations, the instant communications system, station and protocol contemplate first and second oppositely directed signal paths. At least two stations are coupled to both the first and the second signal paths. A station reads one signal from a path and writes another signal

Maguire
maguire@it.kth.se

Patents
2006.03.13

Applications: Network Management and VoIP 547 of 561
Internetworking/Internetteknik

on the path. The one signal is read by an arrangement which electrically precedes the arrangement for writing the other signal. Packets are transmitted in a regular, cyclic sequence. A head station on a forward path writes a start cycle code for enabling each station to transmit one or more packets. If a station has a packet to transmit, it can read the bus field of a packet on the forward path. Responsive thereto, a logical interpretation may be made as to whether the forward path is busy or is not busy. If the path is not busy, the packet may be written on the path by overwriting any signal thereon including the busy field. If the path is busy, the station may defer the writing until the path is detected as not busy. In order to accommodate different types of traffic, the head station may write different start cycle codes. For example, a start-of-voice code may enable stations to transmit voice packets; a start-of-data code may enable stations to transmit data packets, etc. for the different types of traffic. Further, the start cycle codes may be written in a regular, e.g., periodic, fashion to mitigate deleterious effects, such as speech clipping. Still further, the last station on the forward path may write end cycle codes in packets on a reverse path for communicating control information to the head station. Responsive to the control information, the head station may modify the cycle to permit the respective stations to, for example, transmit more than one packet per cycle or to vary the number of packet time slots, which are allocated to each of the different types of traffic.

Maguire
maguire@it.kth.se
Patents
2006.03.13
Applications: Network Management and VoIP 548 of 561
Internetworking/Internetteknik

Deregulation ⇒ Trends

- replacing multiplexors with Routers/Switches/… << 1/10 circuit swi. cost
- Standard telco interfaces being replaced by datacom interfaces
- New Alliances
- future developments building on VOIP

  ◆ Fax broadcast, Improved quality of service, Multipoint audio bridging, Text-to-speech conversion and Speech-to-Text conversion, Voice response systems, …

  ◆ Replacing the wireless voice network's infrastructure with IP

See the Univ. of California at Berkeley ICEBERG project report:

http://iceberg.cs.berkeley.edu/release/

⇒Telecom (only) operators have no future
⇒Telecom (only) companies have no future

Maguire
maguire@it.kth.se
Deregulation ⇒ Trends Applications: Network Management and VoIP 549 of
2006.03.13
Internetworking/Internetteknik

VoIP details

Carry the speech frame inside an RTP packet

IPv4/6	UDP	RTP	CODEC info

20/40 octets   8 octets   12 octets

←————————————————→   ←————————→

40/60 octets   ~33 octets
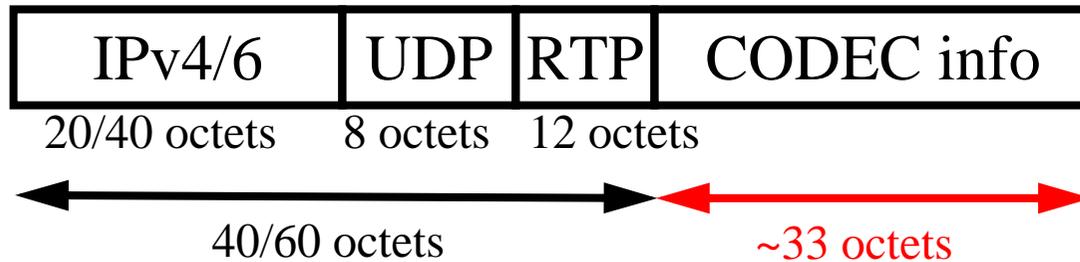
Typical packetisation time of 10-20ms per audio frame.

See `http://www.ietf.org/ids.by.wg/avt.html`

RTP: Real-Time Transport Protocol

Designed to carry out variety of real-time data: audio and video.

Provides two key facilities:

- Sequence number for order of delivery
- Timestamp for control playback

Provides no mechanisms to ensure timely delivery.

0 1	2	3	4	8	9	16
VER	P	X	CC	M	PTYPE	Sequence number
Timestamp						
Synchronization source identifier						
Contributing source ID						
…						
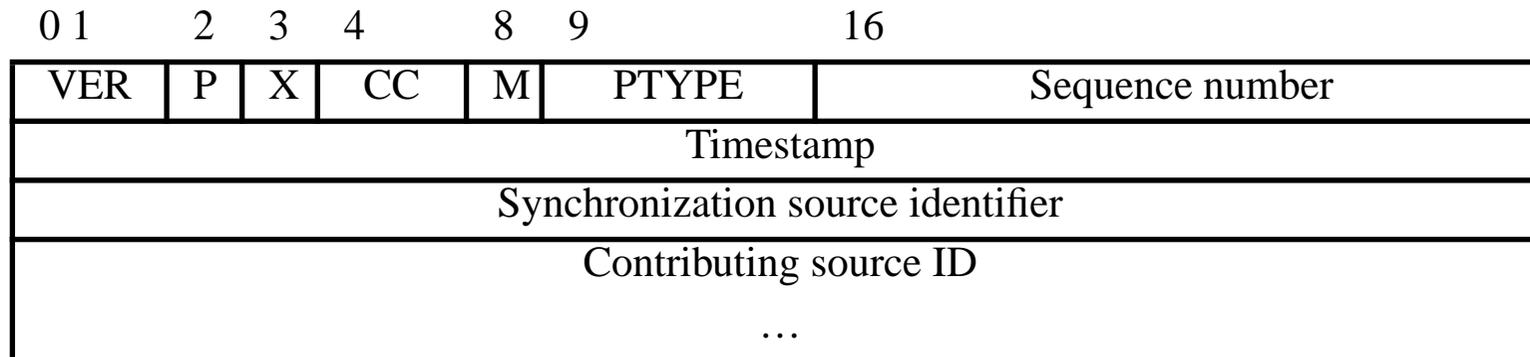
- P - whether zero padding follows the payload.
- X - whether extension or not.
- M - marker for beginning of each frame.
- PTYPE - Type of payload.

RTP and H.323 for IP Telephony

audio/video applications		signaling and control				data applications
video code	audio codec	RTCP	H.225 registration	H.225 Signaling	H.245 Control	T.120
RTP						
UDP				TCP		
IP						

- H.323 is the framework of a group protocols for IP telephony (from ITU)
- H.225 - Signaling used to establish a call
- H.245 - Control and feedback during the call
- T.120 - Exchange of data associated with a call
- RTP   - Real-time data transfer
- RTCP - Real-time Control Protocol

Maguire
maguire@it.kth.se

RTP and H.323 for IP Telephony
2006.03.13

Applications: Network Management and VoIP
Internetworking/Internetteknik

SIP: Session Initiation Protocol

SIP is an alternative to H.323 proposed by IETF. Only covers signaling (parts of H.323). Does not use RTP (but sessions can use RTP)

Several types of servers defined:

- User agent server runs on a SIP terminal = a client element, User Agent Client (UAC) + server element, User Agent Server (UAS)
- SIP proxy - interprets, and, if necessary, rewrites specific parts of a request message before forwarding it to a server closer to the destination:
  - SIP stateful proxy server - remembers its queries and answer; can also forward several queries in parallel.
  - SIP stateless proxy server
- SIP redirect server - directes the client to contact an alternate URI
- Location server - knows the current binding (from REGISTER msgs)

SIP uses SDP (Session Description Protocol) to get information about a call, such as, the media encoding, protocol port number, multicast addresses, etc.

Maguire
maguire@it.kth.se

SIP: Session Initiation Protocol
2006.03.13

Applications: Network Management and VoIP
Internetworking/Internetteknik

SIP timeline

Alice invites Bob to a SIP session:
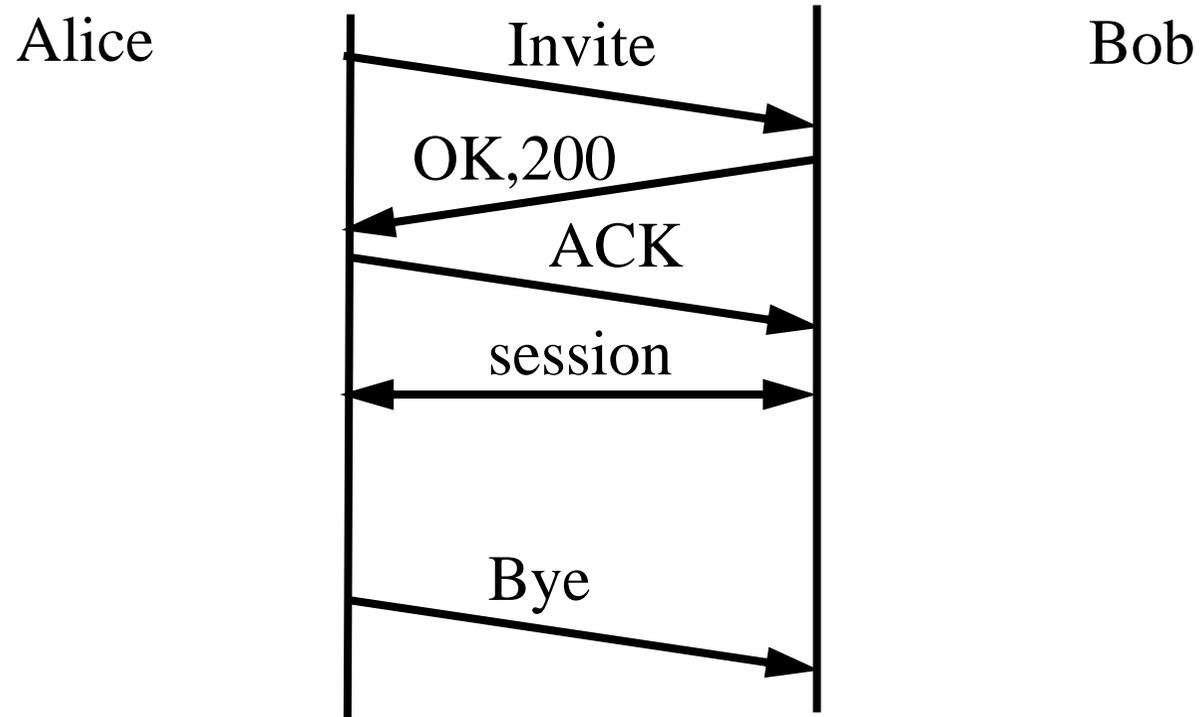
SIP Invite[1]

```
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bK776asdhds
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

(Alices SDP not shown)
```

SIP is a text-based protocol and uses ISO 10646 character set in UTF-8 encoding (RFC 2279). The message body uses MIME and *can* use S/MIME for security.

The generic form of a message is:

```
generic-message = start-line
                  message-header*
                  CRLF
                  [ message-body ]
```

---

1. Example from draft-ietf-sip-rfc2543bis-06.ps

Bob's response[1]

```
SIP/2.0 200 OK Via: SIP/2.0/UDP
pc33.atlanta.com;branch=z9hG4bK776asdhds
Via: SIP/2.0/UDP
bigbox3.site3.atlanta.com;branch=z9hG4bK77ef4c2312983.1
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:bob@192.0.2.8>
Content-Type: application/sdp
Content-Length: 131

(Bobs SDP not shown)
```

---

1. Example from draft-ietf-sip-rfc2543bis-06.ps

SIP Methods

Method	Purpose
Invite	Invites a user to join a call.
Bye	Terminates the call between two of the users on a call.
Options	Requests information on the capabilities of a server.
Ack	Confirms that a client has received a final response to an INVITE.
Register	Provides the map for address resolution, this lets a server know the location of a user.
Cancel	Ends a pending request, but does not end the call.

SIP Status codes

SIP status codes are patterned on and simular to HTTP's status codes:

1xx	Provisional request received, continuing to process the request
2xx	Success - the action was successfully received, understood, and accepted
3xx	Redirection - further action needs to be taken in order to complete the request
4xx	Client Error - the request contains bad syntax or cannot be fulfilled at this server
5xx	Server Error - the server failed to fulfill an apparently valid request
6xx	Global Failure - the request cannot be fulfilled at any server

ENUM

IETF's E.164 Number Mapping standard uses Domain Name Server (DNS) to map standard International Telecommunication Union (ITU-T) international public telecommunications numbering plan (E.164) telephone numbers to a list of Universal Resource Locators (URL). SIP then uses those URL's to initiate sessions.

For example, ENUM DNS converts a telephone number in E.164 format, e.g. +46812345, and returns e.g., a Universal Resource Identifier (URI) SIP:olle.svenson@telia.se

Then a SIP client can make a connection to the SIP gateway telia.se passing the local part olle.svenson.

ENUM can return a wide variety of URI types.

Maguire
maguire@it.kth.se

ENUM
2006.03.13

Applications: Network Management and VoIP 559 of 561
Internetworking/Internetteknik

Further Reading

IP Telephony (*iptel*)

PSTN and Internet Internetworking (*pint*)

Also important are the measures of delay, delay jitter, throughput, packet loss, etc. IP Performance Metrics (*ippm*) is attempting to specify how to measure and exchange information about measurements of these quantities.

A great set of references compiled by prof. Raj Jain is available at:
http://www.cis.ohio-state.edu/~jain/refs/ref_voip.htm

Maguire
maguire@it.kth.se
Further ReadingApplications: Network Management and VoIP 560 of 561
2006.03.13
Internetworking/Internetteknik

Summary

This lecture we have discussed:

- Network Management
- SNMP
- VoIP (including RTP)

Maguire
maguire@it.kth.se

Summary
2006.03.13

Applications: Network Management and VoIP 561 of 561
Internetworking/Internetteknik

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 10: IPv6

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapter 27

Lecture 6: Outline

- IPv6

Maguire
maguire@it.kth.se

Lecture 6: Outline
2006.03.13

IPv6 563 of 613
Internetworking/Internetteknik

Internet Protocol Version 6 (IPv6)

- Successor of current IPv4
- Internet needs to change IP in order to continue growth
- Defines a transition from IPv4 to IPv6

Specified by RFC 2460: Internet Protocol, Version 6 (IPv6) Specification, December 1998[79].

Maguire
maguire@it.kth.se

Internet Protocol Version 6 (IPv6)
2006.03.13

IPv6 564 of 613
Internetworking/Internetteknik

Growth

- Currently IPv4 serves a market doubling every ~12 months
- In addition, new and very large markets are developing rapidly:
  - Nomadic Computing
  - Networked Entertainment
  - Device Control

Maguire
maguire@it.kth.se

Growth
2006.03.13

IPv6 565 of 613
Internetworking/Internetteknik

Nomadic Computing

- Wireless computers
  - supporting multimedia
  - replacing pagers, cellular telephones, …

- IPv6 includes support for mobility
  - low overhead (?)
  - auto configuration
  - mobility

Maguire
maguire@it.kth.se

Nomadic Computing
2006.03.13

IPv6 566 of 613
Internetworking/Internetteknik

Networked Entertainment

Your TV will be an Internet Host!

[consider the network attached Personal Video Recorders (PVR), such as TiVo's DVR, SONICblue's ReplayTV, Sony's SVR-2000, Philips' PTR, …)]

- 500 channels of television
- large scale routing and addressing
- auto-configuration
- requires support for real-time data

SonicBlues's ReplayTV 4000 a networked Digital Video Recorder (DVR) {i.e., coder/decoder + very big disk) that takes advantage of your broadband Internet connection - enables you to capture and transfer videos.

Providing "narrowcast" content via broadband $\Rightarrow$ all the time is "primetime".

Maguire
maguire@it.kth.se

Networked Entertainment
2006.03.13

IPv6 567 of 613
Internetworking/Internetteknik

Device Control

- Control everyday devices for
  - lightning, heating and cooling, motors, ...
  - new street light controllers already have IP addresses!
  - electrical outlets with addresses
  - networked vehicles (within the vehicle[1], between vehicles, and vehicles to infrastructure)[2]
- Market size is enormous
- Solution must be
  - simple, robust, easy to use
  - very low cost
  - potential power savings by (remote) network management based control may be quite large

There is already a networked: Toaster, a Coke machine, … .

---

1. On-Board Diagnostic systems (OBD-II), see slide 8 [80]

2. See InternetCAR, slide 4 (showing a Yokohama City bus) [80]

Maguire
maguire@it.kth.se

Device Control
2006.03.13

IPv6 568 of 613
Internetworking/Internetteknik

IPv6 features

- Expanded Addressing Capabilities
  - 128 bit address length
  - supports more levels of hierarchy
  - improved multicast routing by using a scope field
  - new cluster addresses to identify topological regions

- Header Format Simplification
  - some IPv4 fields have been dropped, some made optional
  - header is easier to compute

- Improved Support for Extensions and Options
  - more efficient for forwarding of packets
  - less stringent limits to length of options
  - greater flexibility for introduction of future options

- Flow Labeling Capability
  - labeling of packets belonging to a particular "flow"
  - allows special handling of, e.g., real-time, packets

- Authentication and Privacy Capabilities
  - Extensions to support authentication, data integrity, and (optional) data confidentiality

Maguire
maguire@it.kth.se

IPv6 features
2006.03.13

IPv6 569 of 613
Internetworking/Internetteknik

IPv6 header format

version 4 bits	Class 8 bits	flow label 20-bits		
"payload" length (in octets) 16 bit			next header 8 bits	hop limit 8 bits
Source Address 128 bits				
Destination Address 128 bits				

IPv6 header (total length = 40 bytes)

IPv6: 6 fields + 2 addresses

   versus

IPv4: 10 fixed fields + 2 addresses + options

Maguire
maguire@it.kth.se

IPv6 header format
2006.03.13

IPv6 570 of 613
Internetworking/Internetteknik

Demultiplexing

Initially, it was assumed that by keeping the version field the same that IPv4 and IPv6 could be mixed over the same links with the same link drivers.

However, now IPv6 will be demultiplexed at the link layer:
hence, IPv6 been assigned the Ethernet type 0x86DD (instead of IPv4's 0x8000)

Maguire
maguire@it.kth.se

Demultiplexing
2006.03.13

IPv6 571 of 613
Internetworking/Internetteknik

Simplifications

IPv6 builds on 20 years of internetworking experience - which lead to the following simplifications and benefits:

Simplification	Benefits
Use fixed format headers	Use extension headers instead, thus no need for a header length field, simpler to process
Eliminate header checksum	Eliminate need for recomputation of checksum at each hop (relies on link layer or higher layers to check the integrity of what is delivered)
Avoid hop-by-hop segmentation	No segmentation, thus you must do **Path MTU discovery** or only send small packets (1996: 536 octets, 1997: proposed 1500 octets) (for observed PMTUs see [82]) • This is because we should have units of **control** based on the units of **transmitted** data.
Eliminate Type of Service (ToS) field	Instead use (labeled) flows

Maguire
maguire@it.kth.se

Simplifications
2006.03.13

IPv6 572 of 613
Internetworking/Internetteknik

Quality-of-Service Capabilities

- for packet streams
- Flow characterized by flow id + source address + destination address
- unique **random** flow id for each source

CLASS (8 bits)	FLOW ID (20 bits)

- Class field

D (1 bit)	Network-wide priority (3 bits)	Reserved (4 bits)
Delay sensitive	Encodes the priority of traffic, can be used to provide "Differentiated services"	Researchers would like to use two of these bits for congestion avoidance control: ♦ one bit which could be set by routers to indicate that congestion was experienced; ♦ the other bit could be used by the source to mark that it is "ready to adapt".

- Flow ID - indicates packets which should all be handled the same way.

The original specified in *RFC 1809: Using the Flow Label Field in IPv6* Subsequently updated - see Chapter 6 of Huitema, **2nd edition**; this change occurred because of Steve McCanne's SigComm'96 paper [84]. **Note that chapter 27 in Forouzan is incorrect**!

Maguire
maguire@it.kth.se

Quality-of-Service Capabilities
2006.03.13

IPv6 573 of 613
Internetworking/Internetteknik

Payload length

Payload length is the length of the data carried after the header.

As the length field is 16 bits $\Rightarrow$ maximum packet size of 64 kilobytes; but there is a provision for "jumbograms" [via the Hop-by-Hop option header with option type 194]. See RFC 2675 [85].

Maguire
maguire@it.kth.se

Payload length
2006.03.13

IPv6 574 of 613
Internetworking/Internetteknik

IPv4 Protocol type ⟹ IPv6 Next Header type

Tells how to interpret the next header which follows, it is either the payload type or the type of the next header. [Payload types use the IPv4 protocol type values]

Decimal	Keyword	Header type
0	HBH	Hop-by-hop options
2	ICMP	IPv6 ICMP
3	GGP	Gateway-to-Gateway Protocol
5	ST	Stream
6	TCP	Transmission Control Protocol
17	UDP	User Datagram Protocol
43	RH	IPv6 Routing Header
44	FH	IPv6 Fragmentation Header
45	IDRP	Inter-domain Routing Protocol
51	AH	Authentication Header
52	ESP	Encrypted Security Payload
59	Null	No next Header (IPv6)
60		IPv6 Destination Options Header
88	IGRP	IGRP
89	OSPF	Open Shortest Path First
255		Reserved

Maguire
maguire@it.kth.se

IPv4 Protocol type ⟹ IPv6 Next Header type
2006.03.13

IPv6 575 of 613
Internetworking/Internetteknik

Extension headers

- Each header is a multiple of 8 octets long

- order (after IPv6 header):
  - Hop-by-hop option,
  - Destination options header (1)
  - Routing header,
  - Fragment header,
  - Authentication header,
  - Encapsulating security payload header,
  - Destination options header (2)
  - Followed by the upper layer header (e.g., TCP, UDP, …)

So a TCP packet looks like:

IPv6 header Next Header = **TCP**	TCP header + data
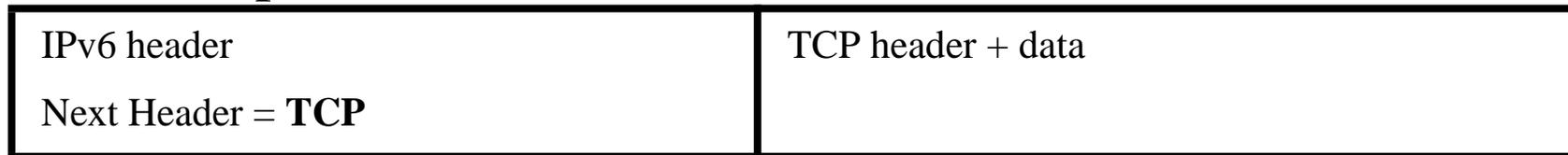
If we wanted to explicitly route the above packet, we simply add a routing header:

IPv6 header Next Header = **Routing**	Routing header Next Header = **TCP**	TCP header + data

Maguire
maguire@it.kth.se

Extension headers
2006.03.13

IPv6 576 of 613
Internetworking/Internetteknik
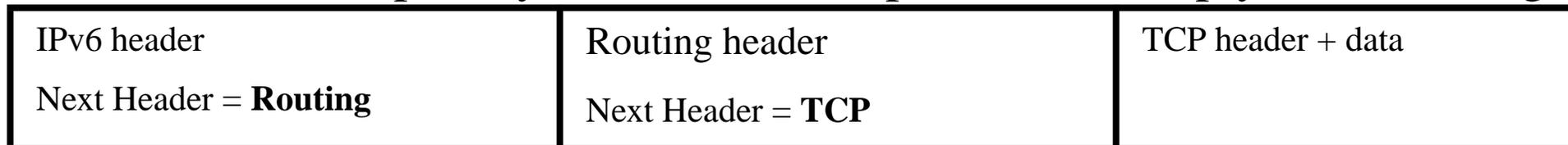
Addressing

- 128 bits long
- three types: unicast, multicast, anycast

Unicast	identifies exactly one interface
Multicast	identifies a group of interfaces; a packet sent to a multicast address will be delivered to all members of the group
Anycast	delivered to the nearest member of the group

- 2^{96} times more addresses than IPv4 are available !!!

IPv6 addresses per m^2

Earth: 511,263,971,197,990 m^2
$\Rightarrow$ 665,570,793,348,866,943,898,599 / m^2

- pessimistic estimate with hierarchies: ~1,564 addresses / m^2
- optimistic: 3,911,873,538,269,506,102 /m^2

Maguire
maguire@it.kth.se

Addressing
2006.03.13

IPv6 577 of 613
Internetworking/Internetteknik

Writing an IPv6 address

The 128 bit IPv6 address is written as eight 16 bit integers using hexadecimal digits.

The integers are separated by colons, for example:
2001:0DB8:7654:3210:FEDC:BA98:7654:3210

A number of abbreviations are allowed:

- leading zeros in integers can be suppressed
- a single set of consecutive 16 bit integers with the value null, can be replaced by double colon, i.e.,
  2001:DB8:0:0:0:0:7654:3210 becomes 2001:DB8::7654:3210
- When an IPv4 address is turned into an IPv6 address we prepend 96 bits of zero; but we can write it as:
  ::10.0.0.1   - hence combining dotted-decimal and IPv6 forms
- Prefixes can be denoted in the same manner as for IPv4, i.e., CIDR:
  2001:DB8::/32   - for a 32 bit long prefix

Maguire
maguire@it.kth.se

Writing an IPv6 address
2006.03.13

IPv6 578 of 613
Internetworking/Internetteknik

Address Allocation [95] and [100]

Binary prefix	Hex. prefix	Fraction of address space	Assignment
0000 0000	::/8	1/256	Reserved
0000 0001	100::/8	1/256	Unassigned
0000 001	200::/7	1/128	Network Service Access Point (NSAP) Allocation-RFC 1888
0000 01	400::/6	1/64	Unassigned (first half was formerly Novell's IPX)
0000 1	800::/5	1/32	Unassigned
0001	1000::/4	1/16	Unassigned
001	**2000::/3**	**1/8**	**Global Unicast - RFC 2374 see RFC 3587**
010	4000::/3	1/8	Unassigned (formerly provider based unicast addresses)
011	6000::/3	1/8	Unassigned
100	8000::/3	1/8	Unassigned (formerly Geographic-based Unicast Addresses)
101	A000::/3	1/8	Unassigned
110	C000::/3	1/8	Unassigned
1110	E000::/4	1/16	Unassigned
1111 0	F000::/5	1/32	Unassigned
1111 10	F800::/6	1/64	Unassigned
1111 110	FC00::/7	1/128	Unassigned
1111 1110 0	FE00::/9	1/512	Unassigned
1111 1110 10	**FE80::/10**	**1/1024**	**Link Local Use Addresses**
1111 1110 11	**FEC0::/10**	**1/1024**	**Reserved for IANA (was Site Local Use Addresses)**
1111 1111	**FF00::/8**	**1/256**	**Multicast Addresses**

Maguire
maguire@it.kth.se

Address Allocation [95] and [100]
2006.03.13

IPv6 579 of 613
Internetworking/Internetteknik

Global Unicast Addresses

RFC 2374 defined an IPv6 address allocation structure that which featured Top Level Aggregator (TLAs) and Next Level Aggregator (NLAs) - this has been replaced (see RFC 3587[96]) by a coordinated allocation policy defined by the Regional Internet Registries (RIRs) [97]

The Subnet Local Aggregator (SLAs) of RFC 2374 $\Rightarrow$ now called the "subnet ID"

001 (3 bits)	global routing prefix (45 bits)	subnet ID (16 bits)	Interface ID (64 bits)

Thus the Regional Internet Registries are allocating addresses from 2000:/3

For a table of IPv6 unicast assignment see

http://www.iana.org/assignments/ipv6-unicast-address-assignments

For an analysis of use from the point of view of RIPE see [102]

Maguire
maguire@it.kth.se

Global Unicast Addresses
2006.03.13

IPv6 580 of 613
Internetworking/Internetteknik

Interface ID

Must be unique to the link, but there are some advantages of making it more globally unique.

Hence, most will be based on the IEEE EUI-64 format, but with the "u" (unique) bit inverted.

- The "u" bit is the 7th most significant bit of a 64 bit EUI.
- The inversion was necessary because 0:0:0:0 is a valid EUI, but this would collide with one of the IPv6 special addresses.
- u=1, when the address comes from a valid EUI, and is 0 otherwise.

To go from a 48 bit IEEE 802, you insert 0xFFFE in between the 3rd and 4th octets of an IEEE 802 address, i.e., 123456789abc becomes 123456FFFE789abc.

Maguire
maguire@it.kth.se

Interface ID
2006.03.13

IPv6 581 of 613
Internetworking/Internetteknik

Special Address Formats

Unspecified address

"::" == "0:0:0:0:0:0:0:0 - can only be used as a source address by a station which does not yet have an address

Loop-back address

0:0:0:0:0:0:0:1 - used to send an IPv6 datagram to yourself

IPv4-based address

prefix the 32 bit IPv4 address with 96 zero bits

Site local addresses

Site local address can not be routed on the global internet, but they can be used by sites that are not connected to the internet or for communication within the site.

1111111011 (10 bits)	0 (38 bits)	Subnet (16 bits)	Interface ID (64 bits)

Maguire
maguire@it.kth.se

Special Address Formats
2006.03.13

IPv6 582 of 613
Internetworking/Internetteknik

Link local addresses

Link local address are simply unique to a given link - they can be used by stations that have not yet been assigned a provider-based address.

1111111010 (10 bits)	0 (54 bits)	Interface ID (64 bits)

Maguire
maguire@it.kth.se

Special Address Formats
2006.03.13

IPv6 583 of 613
Internetworking/Internetteknik

Multicast Addresses

	Flags	Scope	112 bit - group id
1111 1111	Flags xxxT	Scope	112 bit - group id

4 bit   4 bit

T == Transient

T = 0	well-known permanent - assigned by the IANA
T = 1	non-permanent

Scope

0	reserved
1	node local scope
2	link local scope
3, 4	unassigned
5	site local scope
6, 7	unassigned
8	organization local scope
9, A, B, C, D	unassigned
E	global scope
F	reserved

Maguire
maguire@it.kth.se

Multicast Addresses
2006.03.13

IPv6 584 of 613
Internetworking/Internetteknik

Permanently assigned groups

For example, group 0x43 has been assigned to the Network Time Protocol (NTP), hence:

FF01::43	represents all NTP servers on the same node as the sender
FF02::43	represents all NTP servers on the same link as the sender
FF05::43	represents all NTP servers on the same site as the sender
FF08::43	represents all NTP servers within the same organization as the sender
FF0E::43	represents all NTP servers in the Internet

IANA has assigned a whole series of group identifiers, including:

FF0X:0:0:0:0:0:0	Reserved multicast address - this can not be used within any scope
FF01:0:0:0:0:0:1	All Nodes on this node address
FF02:0:0:0:0:0:1	All Nodes on this link address
FF01:0:0:0:0:0:2	All Routers on this node address
FF02:0:0:0:0:0:2	All Router address on this link

Maguire
maguire@it.kth.se

Permanently assigned groups
2006.03.13

IPv6 585 of 613
Internetworking/Internetteknik

FF02:0:0:0:0:0:3	unassigned
FF02:0:0:0:0:1:1	Link Name
FF02:0:0:0:0:1:2	All DHCP agents on this link
FF02:0:0:0:0:1:3	All DHCP servers on this link
FF02:0:0:0:0:1:4	All DHCP relays on this link
FF05:0:0:0:0:1:2	All DHCP agents at this site
FF05:0:0:0:0:1:3	All DHCP servers at this site
FF05:0:0:0:0:1:4	All DHCP relays at this site
FF0X:0:0:0:0:2:7FFE	Session Announcement Protocol (SAP) v1 Announcements

Multimedia conferences:

FF0X:0:0:0:0:2:8000 .. FF0X:0:0:0:0:2:FFFF	multimedia conferences

X=2 -- this link; X=5 -- this site

Use SAP to announce the conference - repeatedly until the end of the conference.

Maguire
maguire@it.kth.se

Permanently assigned groups
2006.03.13

IPv6 586 of 613
Internetworking/Internetteknik

Prefix for IPv6 documentation

The IPv6 unicast address prefix `2001:DB8::/32` is reserved for use in examples for books, RFCs, … [86].

Note that this is a **non-routable** range - to help avoid problems!

Maguire
maguire@it.kth.se

Prefix for IPv6 documentation
2006.03.13

IPv6 587 of 613
Internetworking/Internetteknik

Anycast

Sending a packet to a generic address to get a specific service from the "nearest" instance. This puts the burden of determining which instance to deliver it to on the routing system.

Requires defining a router entry for each anycast address.

Subnet Anycast Address:

Subnet prefix (n bits)	0 (128-n bits)

Thus the host ID of zero is treated as the subnet.

Maguire
maguire@it.kth.se

Anycast
2006.03.13

IPv6 588 of 613
Internetworking/Internetteknik

IPv6 Routing

- all standard routing protocols

- routing extensions
  - Provider Selection
  - Host Mobility (route to current location)
  - Auto-Readdressing (route to new address)
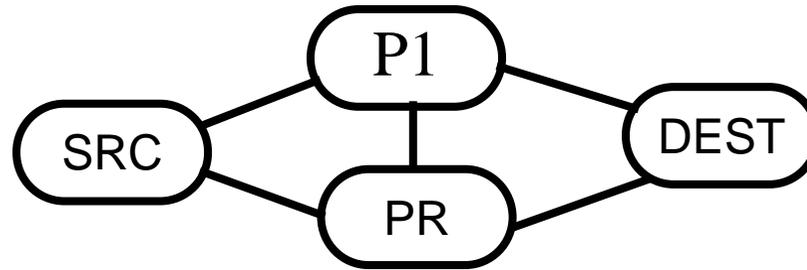
Figure 92: IPv6 Routing Option: provider specifies: SRC, PR, P1, Dest

reply: Dest, PR, P1, SRC

Maguire
maguire@it.kth.se

IPv6 Routing
2006.03.13

IPv6 589 of 613
Internetworking/Internetteknik

Routing header

Next Header (8 bits)	Header Ext Length (8 bits)	Routing Type=0 (8 bits)	Segments Left (8 bits)
reserved (32 bits)			
address[1] (128 bits)			
address[2]			
...			
address[n]			

Next Header identifies the next header in the chain of headers.

Header Ext. Length. - number of 64 bit words (not including the first 64 bits).

Routing type=0, is the generic routing header which all IPv6 implementations must support.

Number of Segments is the number of segments left in the list (between 0 and 23).

Maguire
maguire@it.kth.se

Routing header
2006.03.13

IPv6 590 of 613
Internetworking/Internetteknik

Fragment header

Next Header (8 bits)	Reserved (8 bits)	Fragment offset (13 bits)	RESERVED (2 bits)	M (1 bit)
Identification				

Fragment offset - in units of 64 bit words, the field is the most significant 13 bits of a 16 bit words.

M == More fragment bit, set in all but the last fragment

Identification - a 32 bit number
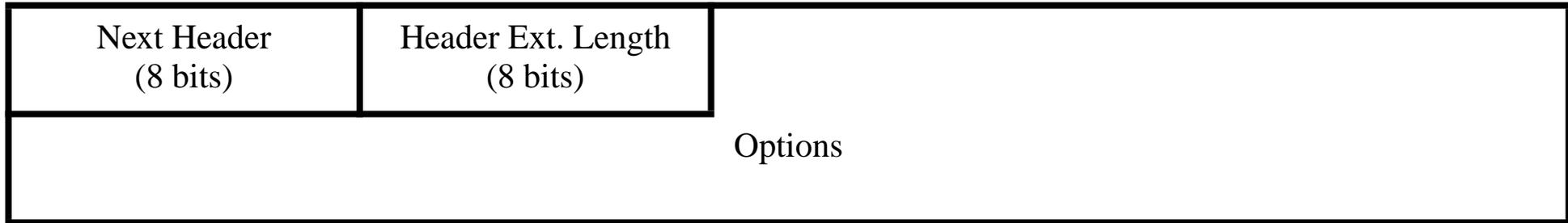
Maguire
maguire@it.kth.se

Fragment header
2006.03.13

IPv6 591 of 613
Internetworking/Internetteknik

Destination Options header

Next Header (8 bits)	Header Ext. Length (8 bits)	
Options		

Each options field is encoded as:

Option Type (8 bits)   Option Data Length (8 bits)   Option Data (n octets)

The option type:

Action (2 bits)   C (1 bit)   Number (6 bits)

Action tells what action must be taken if the processing nodes does **not** recognize the option.

Bits	Action
00	Skip over this option
01	Discard packet silently (i.e., without sending an ICMP report)
10	Discard packet and send an ICMP report - even if destination is multicast
11	Discard packet and send an ICMP report - only if destination is **not** multicast

C == change en route bit -- indicates that this option may be changed by

Maguire
maguire@it.kth.se

Destination Options header
2006.03.13

IPv6 592 of 613
Internetworking/Internetteknik

intermediate relays on the way to the destination

Currently only two options are defined:

Pad1 == a null byte - for use in padding to a 64-bit boundary; note it does not have a null option length field after it - as it is the whole field

PadN - the length field says how many null bytes are needed to file to a 64-bit boundary.

Maguire
maguire@it.kth.se
Destination Options header
2006.03.13
IPv6 593 of 613
Internetworking/Internetteknik

Hop-by-Hop Options header

Same basic format as Destination option header, but the hop-by-hop header will be processed at each hop along the way.

Next Header (8 bits)	Header Ext. Length (8 bits)	
Options		

Each options field is encoded as:
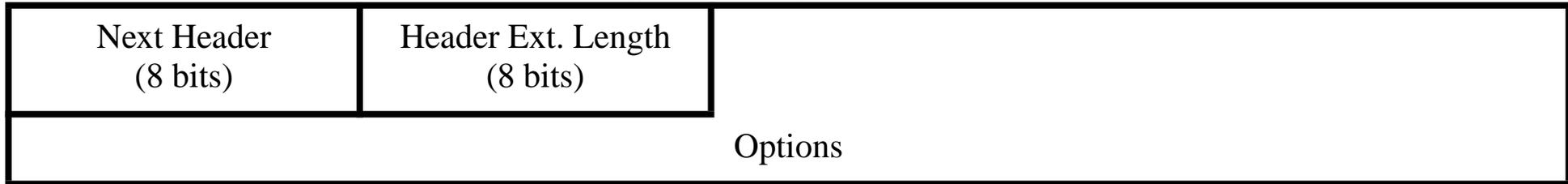
Option Type (8 bits)	Option Data Length (8 bits)	Option Data (n octets)
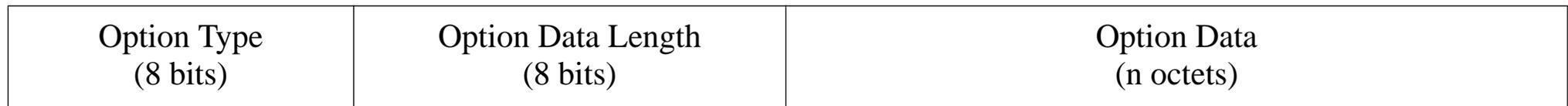
Currently three options are defined: Pad1, PadN, and

- Jumbo payload option (option type =194) - the option Data Length is 4 and is followed by a 32 bit Jumbo Payload Length value.

See RFC 2113: Router Alert Option [87].

Maguire
maguire@it.kth.se

Hop-by-Hop Options header
2006.03.13

IPv6 594 of 613
Internetworking/Internetteknik

Security

- Header Authentication with signatures
  - Must have support for Message Digest 5 (MD5) algorithm [89]
- RFC 1810 [90] examines MD5 performance
- Packet Encapsulation with e.g., DES

For more information see Chapter 5 of *IPv6*, 2nd edition, by Christian Huitema.

Maguire
maguire@it.kth.se

Security
2006.03.13

IPv6 595 of 613
Internetworking/Internetteknik

IPSEC IPv6 implementation

The US Naval Research Lab (NRL) IPv6/IPsec Software Distribution

- a reference implementation of IPv6 and IP Security for the 4.4BSD-Lite networking software.

- Freely distributable (subject to U.S. export controls) and usable for commercial and non-commercial purposes (you must adhere to the NRL and UC Berkeley license terms) see also:
  http://web.mit.edu/network/isakmp
  - DOD ISAKMP Distribution
  - Cisco's ISAKMP Distribution
  - NRL's IPv6 + IPSEC Alpha 7.1 Distribution (Dec '98)
  - Portland State University's Mobile IP with IPSEC for FreeBSD 2.2.1.

See also the list of IPv6 implementations at:

http://playground.sun.com/pub/ipng/html/ipng-implementations.html

Maguire
maguire@it.kth.se

IPSEC IPv6 implementation
2006.03.13

IPv6 596 of 613
Internetworking/Internetteknik

IPv6 ICMP [91]

Type (8 bits)	Code (8 bits)	Checksum (16 bits)
Message Body		

Currently defined ICMP Types

Type	Purpose
1	Destination Unreachable
2	Packet too big
3	Time exceeded
4	Parameter problem
128	Echo Request
129	Echo Reply
130	Group Membership Query
131	Group Membership Report
132	Group Membership Reduction
133	Router Solicitation
134	Router Advertisement
135	Neighbor Solicitation
136	Neighbor Advertisement
137	Redirect

Maguire
maguire@it.kth.se

IPv6 ICMP [91]
2006.03.13

IPv6 597 of 613
Internetworking/Internetteknik

IPv6 ICMP Error Messages

Type: 1, 2, 3, or 4:

Type	Code	Checksum
Parameter		
As much of invoking packet as will fit - without the overall ICMP packet exceeding 576 octets		

For type 1 the code reveals the reason for discarding the datagram

Maguire
maguire@it.kth.se

IPv6 ICMP Error Messages
2006.03.13

IPv6 598 of 613
Internetworking/Internetteknik

IPv6 ICMP Echo Request/Reply (PING)

Type: Echo Request = 128, Echo Reply = 129

Type	Code	Checksum
Identifier		Sequence number
Data		

Maguire
maguire@it.kth.se

IPv6 ICMP Echo Request/Reply (PING)
2006.03.13

IPv6 599 of 613
Internetworking/Internetteknik

IPv6 ICMP and groups

Three group membership messages (type 130, 131, and 132):

Type	Code	Checksum
Maximum Response Delay		Unused
Multicast Address		

The Group Membership Reduction is used when a node leaves group.

Reports are always sent to the same group address that is reported.

Maximum response delay is the time in milliseconds that the responding report messages can be delayed. Responding stations are supposed to spread their responses uniformly over this range of delays (to prevent everyone from responding at once).

Maguire
maguire@it.kth.se

IPv6 ICMP and groups
2006.03.13

IPv6 600 of 613
Internetworking/Internetteknik

Summary of IPv6 ICMP

- incorporates IPv4's ARP (via neighbor solicitation and advertisement) and IGMP (via group membership messages)
- RARP is **dropped** since BOOTP provides the same functionality
- **dropped** IPv4's Source Quench
- added Packet Too Big message to simplify learning MTU size

For more information about ICMP see: RFC 2463 [91]

Maguire
maguire@it.kth.se

Summary of IPv6 ICMP
2006.03.13

IPv6 601 of 613
Internetworking/Internetteknik

DNS and IPv6

A new record type "AAAA" which contains a 128 bit address.

Just as for the "in-addr.arpa" domain used for converting Ipv4 addresses into names, IPv6 defines an "ipv6.int" domain:
thus the address 2001:0DB8:1:2:3:4:567:89ab is represented as:
b.a.9.8.7.6.5.0.4.0.0.0.3.0.0.0.2.0.0.0.1.0.0.0.8.b.d.0.1.0.0.2.IP6.INT

For further information see RFC 3596 [92].

Maguire
maguire@it.kth.se
DNS and IPv6
2006.03.13
IPv6 602 of 613
Internetworking/Internetteknik

IPv6 Transition Mechanisms

- Incremental update and deployment
  - first step: dual stack hosts and routers
  - Encapsulation of IPv6 in IPv4 packets $\Rightarrow$ tunnels
- Minimal upgrade dependencies (must first upgrade DNS)
- Easy addressing (upgraded routers can use IPv4 address)
- FreeBit Co., Ltd.'s Feel6 - secure IPv6 over IPv4[83], see slide 12 [79] and *http://start.feel6.jp/*
- See also [93]

Maguire
maguire@it.kth.se

IPv6 Transition Mechanisms
2006.03.13

IPv6 603 of 613
Internetworking/Internetteknik

Why IPv6?

- solves Internet scaling problem
  - "eliminates" the problem of running out of addresses
  - allows route aggregation - which allows the size of the routing tables in the backbone routers to decrease

- flexible transition (interworks with IPv4)

- meets the needs of new markets

- new functionality

- real-time flows

- provider selection

- host mobility

- end-to-end security

- auto-configuration - chapter 4, "Plug and Play" in *IPv6*, 2nd edition, by Christian Huitema - this a **very major** advantage of IPv6. See also [99]

Maguire
maguire@it.kth.se

Why IPv6?
2006.03.13

IPv6 604 of 613
Internetworking/Internetteknik

IPv6 networks

6Bone - *http://www.6bone.net/* a testbed for deployment of IPv6

- Note the phase out of the 3FFE::/16 prefix
- prefix will be returned to the unassigned address pool on 6 June 2006 [94].

vBNS - *http://www.vbns.net*

6NET *http://www.6net.org/* - project co-funded by European Commission

Euro6IX: European IPv6 Internet Exchanges Backbone

http://www.euro6ix.org/main/index.php - project co-funded by European Commission

For some issues concerning IPv6 deployment see [101]

Maguire
maguire@it.kth.se

IPv6 networks
2006.03.13

IPv6 605 of 613
Internetworking/Internetteknik

RIR assignments of IPv6 addresses

Total number of allocated IPv6 prefixes per RIR on 13/05/2005 [103]

RIR	Size in /48s	Count
APNIC	416,579,590	376
ARIN	9,699,375	195
LACNIC	1,048,577	17
RIPE NCC	1,091,723,264	635
Total:	1,519,050,806	1,223

Maguire
maguire@it.kth.se
RIR assignments of IPv6 addresses
2006.03.13
IPv6 606 of 613
Internetworking/Internetteknik

Further information

See:

http://www.ipv6.org/

http://www.ipv6forum.com/

Measurements of dual stack IPv6 implementations: *http://mawi.wide.ad.jp/mawi/dualstack/*

See also: [81] and [82].

Maguire
maguire@it.kth.se

Further information
2006.03.13

IPv6 607 of 613
Internetworking/Internetteknik

Summary

This lecture we have discussed:

- IPv6

Maguire
maguire@it.kth.se

Summary
2006.03.13

IPv6 608 of 613
Internetworking/Internetteknik

References

[79] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", IETF RFC 2460, December 1998 `http://www.ietf.org/rfc/rfc2460.txt`

[80] Jun Murai, "WIDE report", 5th CAIDA-WIDE Workshop, Information Sciences Institute, Marina del Rey, CA, 15 March 2005

`http://www.caida.org/projects/wide/0503/slides/murai.pdf`

[81] Kenjiro Cho, "Measuring IPv6 Network Quality" (part 1), 5th CAIDA-WIDE Workshop, Information Sciences Institute, Marina del Rey, CA, 15 March 2005 `http://www.caida.org/projects/wide/0503/slides/kenjiro-1.pdf`

[82] Kenjiro Cho, "Measuting IPv6 Network Quality" (part 2), `Internet Iniative Japan (IIJ)` / WIDE, 5th CAIDA-WIDE Workshop, Information Sciences Institute, Marina del Rey, CA, 15 March 2005

`http://www.caida.org/projects/wide/0503/slides/kenjiro-2.pdf`

[83] "Trying Out for Yourself: Smooth use of IPv6 from IPv4 by Feel6 Farm",

Maguire
maguire@it.kth.se

References
2006.03.13

IPv6 609 of 613
Internetworking/Internetteknik

IPv6Style, NTT Communications, 7 March 2003

http://www.ipv6style.jp/en/tryout/20030307/index.shtml

[84] S. McCanne, V. Jacobson, M. and Vetterli, "Receiver-driven Layered Multicast", ACM SIGCOMM, August 1996, Stanford, CA, pp. 117-130.

ftp://ftp.ee.lbl.gov/papers/mccanne-sigcomm96.ps.gz

[85] D. Borman, S. Deering, and R. Hinden, "IPv6 Jumbograms", IETF RFC 2675 August 1999 *http://www.ietf.org/rfc/rfc2675.txt*

[86] G. Huston, A. Lord, and P. Smith, "IPv6 Address Prefix Reserved for Documentation", IETF RFC   3849, July 2004 *http://www.ietf.org/rfc/rfc3849.txt*

[87] Dave Katz, "IPv6 Router Alert Option", IETF RFC 2113, February 1997

http://www.ietf.org/rfc/rfc2113.txt

[88] R. Gilligan,S. Thomson, J. Bound, and W. Stevens, "Basic Socket Interface Extensions for IPv6", IETF RFC 2133, April 1997

http://www.ietf.org/rfc/rfc2133.txt

Maguire
maguire@it.kth.se

References
2006.03.13

IPv6 610 of 613
Internetworking/Internetteknik

[89] R. Rivest, "The MD5 Message-Digest Algorithm", IETF RFC 1321, April 1992 *http://www.ietf.org/rfc/rfc1321.txt*

[90] J. Touch, "Report on MD5 Performance", IETF RFC 1810, June 1995

http://www.ietf.org/rfc/rfc1810.txt

[91] A. Conta and S. Deering, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", IETF RFC 2463, December 1998 *http://www.ietf.org/rfc/rfc2463.txt*

[92] S. Thomson, C. Huitema, V. Ksinant, and M. Souissi, "DNS Extensions to Support IP Version 6", IETF RFC 3596, October 2003

[93] C. Huitema, R. Austein, S. Satapati, and R. van der Pol, "Unmanaged Networks IPv6 Transition Scenarios", IETF RFC 3750 , April 2004

http://www.ietf.org/rfc/rfc3750.txt

[94] R. Fink and R. Hinden, "6bone (IPv6 Testing Address Allocation) Phaseout", IETF RFC 3701, March 2004 *http://www.ietf.org/rfc/rfc3701.txt*

Maguire
maguire@it.kth.se

References
2006.03.13

IPv6 611 of 613
Internetworking/Internetteknik

[95] R. Hinden and S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture", IETF RFC 3513, April 2003 *http://www.ietf.org/rfc/rfc3513.txt*

[96] R. Hinden, S. Deering, and E. Nordmark, "IPv6 Global Unicast Address Format", IETF RFC 3587, August 2003 *http://www.ietf.org/rfc/rfc3587.txt*

[97] APNIC, ARIN, and RIPE NCC, "IPv6 Address Allocation and Assignment Policy", Document ID: ripe-267, January 22, 2003

http://www.ripe.net/ripe/docs/ipv6policy.html

[98] IAB, "IAB/IESG Recommendations on IPv6 Address Allocations to Sites", IETF RFC 3177, September 2001 *http://www.ietf.org/rfc/rfc3177.txt*

[99] S. Thomson and T. Narten, "IPv6 Stateless Address Autoconfiguration", IETF RFC 2462, December 1998

[100] Toshiyuki Hosaka, "IPv6 Address Allocation and Policy: PART1 IPv6 Address Basics", Tech Tutorials, IPv6Style, NTT Communications, 18 November 2004 *http://www.ipv6style.jp/en/tech/20041117/index.shtml*

Maguire
maguire@it.kth.se

References
2006.03.13

IPv6 612 of 613
Internetworking/Internetteknik

[101]Cisco, "IPv6 Deployment Strategies", 23 December 2002 ,

  http://www.cisco.com/univercd/cc/td/doc/cisintwk/intsolns/ipv6_sol/ipv6dswp.htm

[102]Gert Döring, "Impressions:An overview of the global IPv6 routing table", RIPE 50, Stockholm, SE, 3 May 2005, *http://www.space.net/~gert/RIPE/R50-v6-table.pdf*

[103] RIPE, "Total number of allocated IPv6 prefixes per RIR on 13/05/2005", web page accessed 2005.05.14 *http://www.ripe.net/rs/ipv6/stats/index.html*

[104]J. Rajahalme, A. Conta, B. Carpenter, and S. Deering. "IPv6 Flow Label Specification", IETF RFC 3697,   March 2004 *http://www.ietf.org/rfc/rfc3697.txt*

Maguire
maguire@it.kth.se

References
2006.03.13

IPv6 613 of 613
Internetworking/Internetteknik

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 11: Mobile IP

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapter 24

KTH Information and Communication Technology

Outline

- Mobile IP

Maguire
maguire@it.kth.se

Outline
2006.03.13

Mobile IP 615 of 663
Internetworking/Internetteknik

Emerging Network Architecture
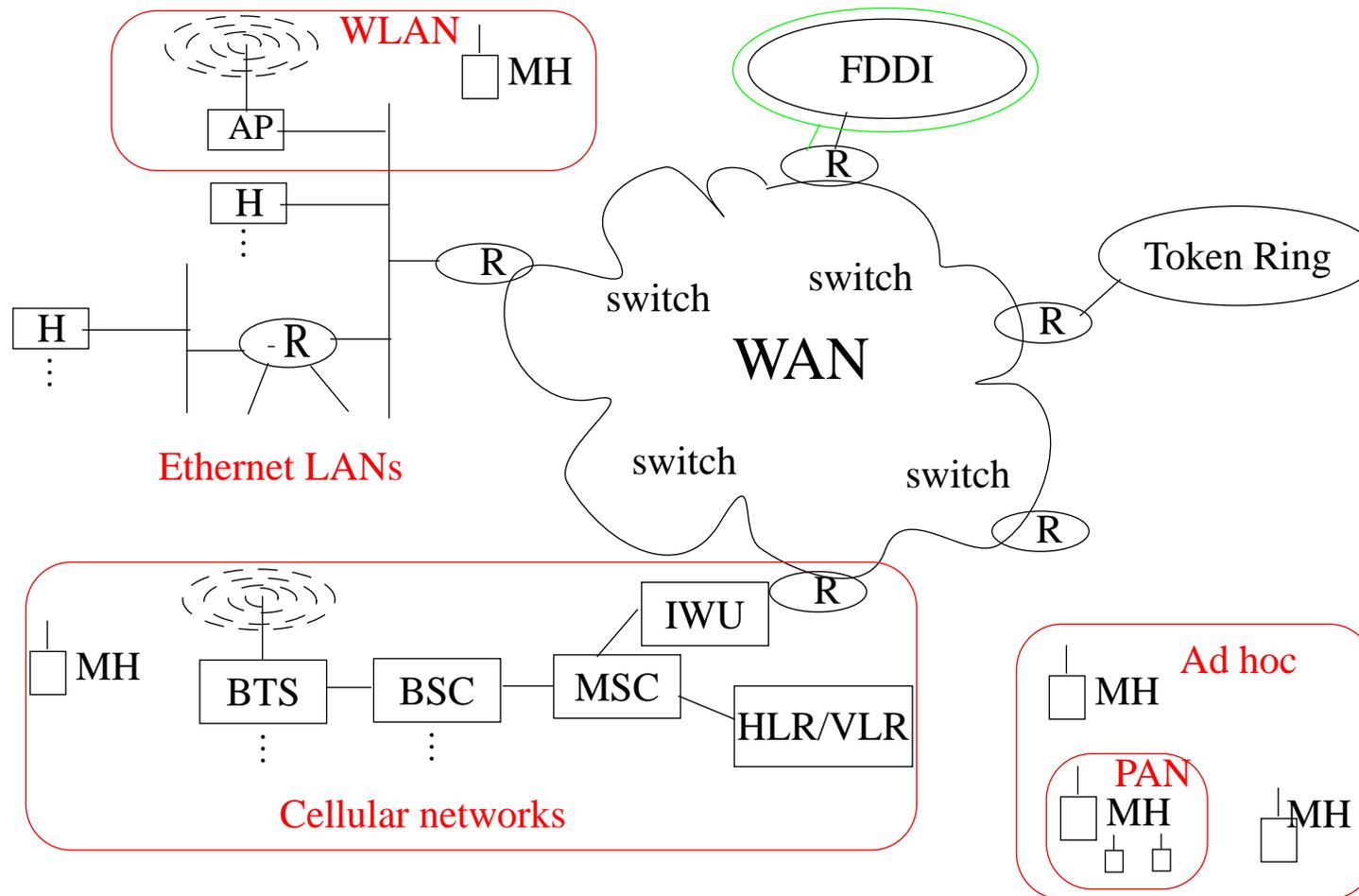
Figure 93: Mobility (WWAN, WLAN, PAN, …) driving us towards Mobile Internet

Maguire
maguire@it.kth.se

Emerging Network Architecture
2006.03.13

Mobile IP 616 of 663
Internetworking/Internetteknik
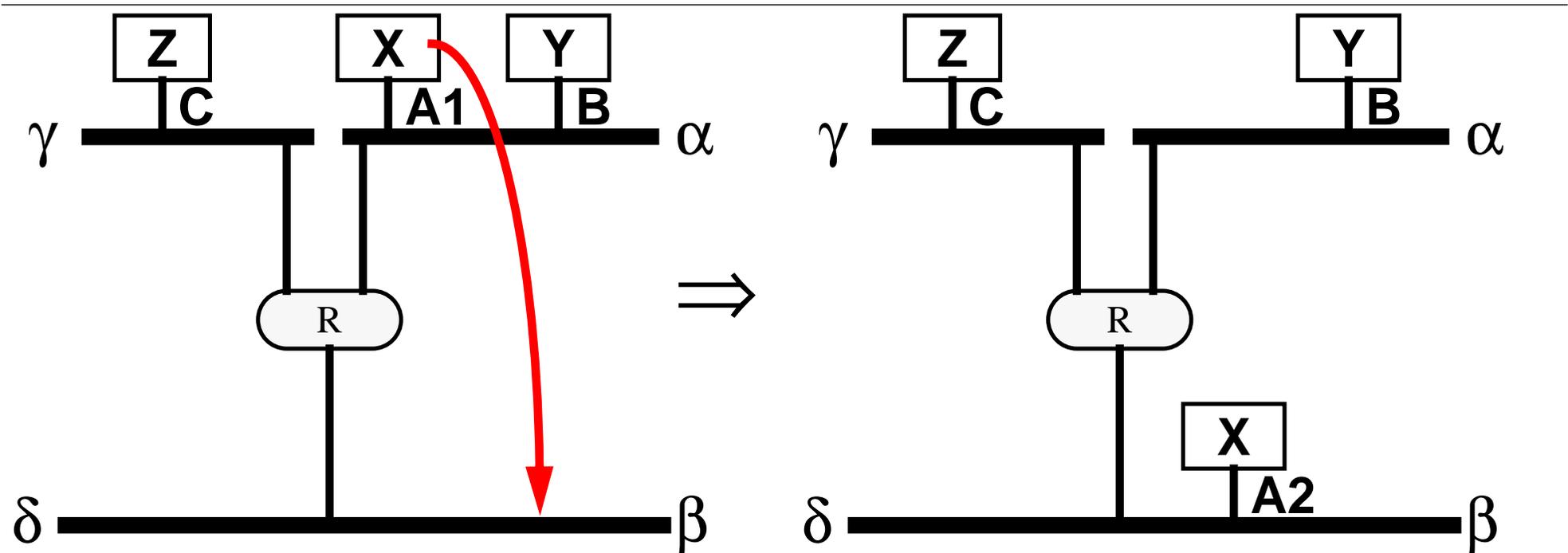
Figure 94. X disconnects from location A1 and reconnects at location A2

What is "X"? X represents the identity (ID) of the node[1]

- in an Ethernet it might be the MAC address, thus a node has a constant identity

While A1, A2, … represent the network addresses of node X.

- IP network address consists of {Network, Host}, i.e., A1={α | n}, where n is unique on network α.

---

1. Of course this really mixes the interface ID with the node ID - solution is a Network Access Identifier[105].

Maguire
maguire@it.kth.se

Emerging Network Architecture
2006.03.13

Mobile IP 617 of 663
Internetworking/Internetteknik

Updating after a move

Host name: **"ccslab1.kth.se"**

Name Resolution: DNS, Host File, … ⇐ DNS, Host File, …

IP address **130.237.15.254** ⇐ **130.237.216.25**

HW address: **Ethernet MAC address** **08:00:2B:00:EE:0B**
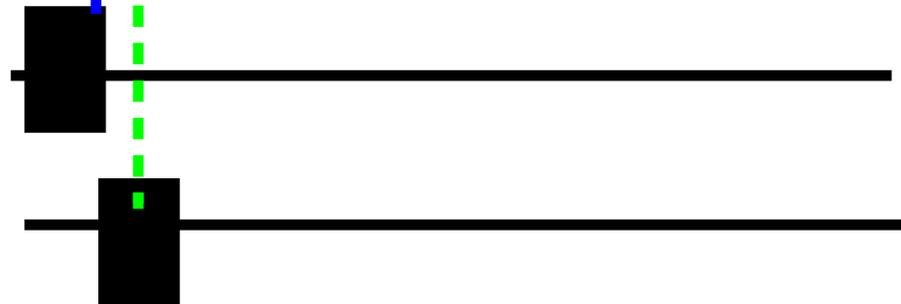
Figure 95: Must update IP address related mappings after a move ⇒ administrative nightmare

Maguire
maguire@it.kth.se

Updating after a move
2006.03.13

Mobile IP 618 of 663
Internetworking/Internetteknik

Objectives of Mobile IP

- To provide mobility support for the Internet
- To enable node mobility: across changes in IP subnet
- Allow change in location without change of IP address
- Communication should be possible (even) while moving (if the interface/link supports it)
- TCP/IP connections should survive movement
- Active TCP and UDP port bindings should be maintained

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 619 of 663
Internetworking/Internetteknik
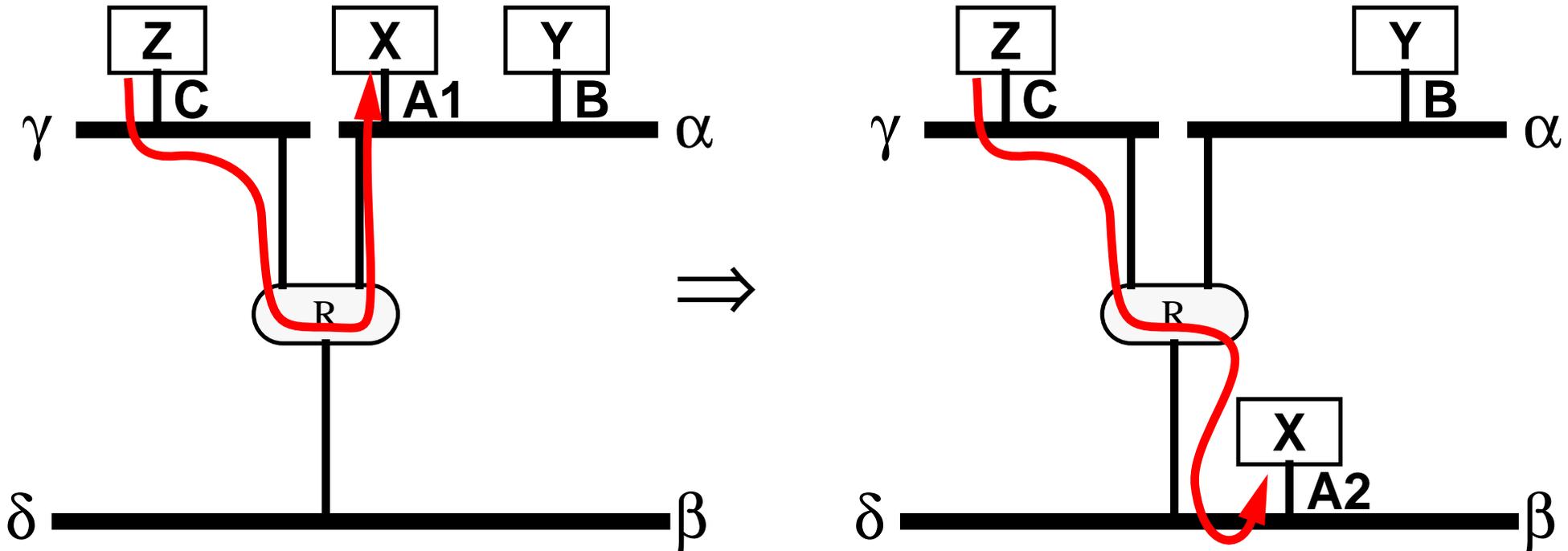
Figure 96. Z is communicating with X at A1 and wants to continue when X reconnects at location A2

- This would require that router R send packets from Z to X over a new path (route).
  - ✘ But X now has a new network address, since it is on a different network (β).

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 620 of 663
Internetworking/Internetteknik

How can Z continue to communication to X?

1. Just use bridging and change the forwarding table in the bridge (since the bridge uses MAC addresses)

   ✘ But bridging does **not** scale well

2. The application could stop, then restart with the new address for X

   ✘ This is unpleasant for the user - since they might have to do this very frequently and/or the programs may not tolerate this change - since they have too much state.

3. We could hide this change with a new layer of software

   a. We could change the socket library

   ✘ for example: we could do source routing - but, it turns out that this is not well supported by existing code in the OS[1] and in router (in addition, many the firewall routers at many sites filter out source routed packets!)

   ✘ Would require changes in all systems (even the non-mobile systems - since both ends of the communication would have to change)

   b. We could remap the addresses in the router

   ✘ This would means doing host specific routing, which does not scale well

   c. We could define a new Mobile-IP address

   ✔ The implications of this will be described in the following material.

---

1. An informal experiment conduced by John Ioannidis as part of this Mobile*IP research (and documented in an appendix of his thesis) indicted that almost all operating systems, of the time, did not correctly support source routing!

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 621 of 663
Internetworking/Internetteknik
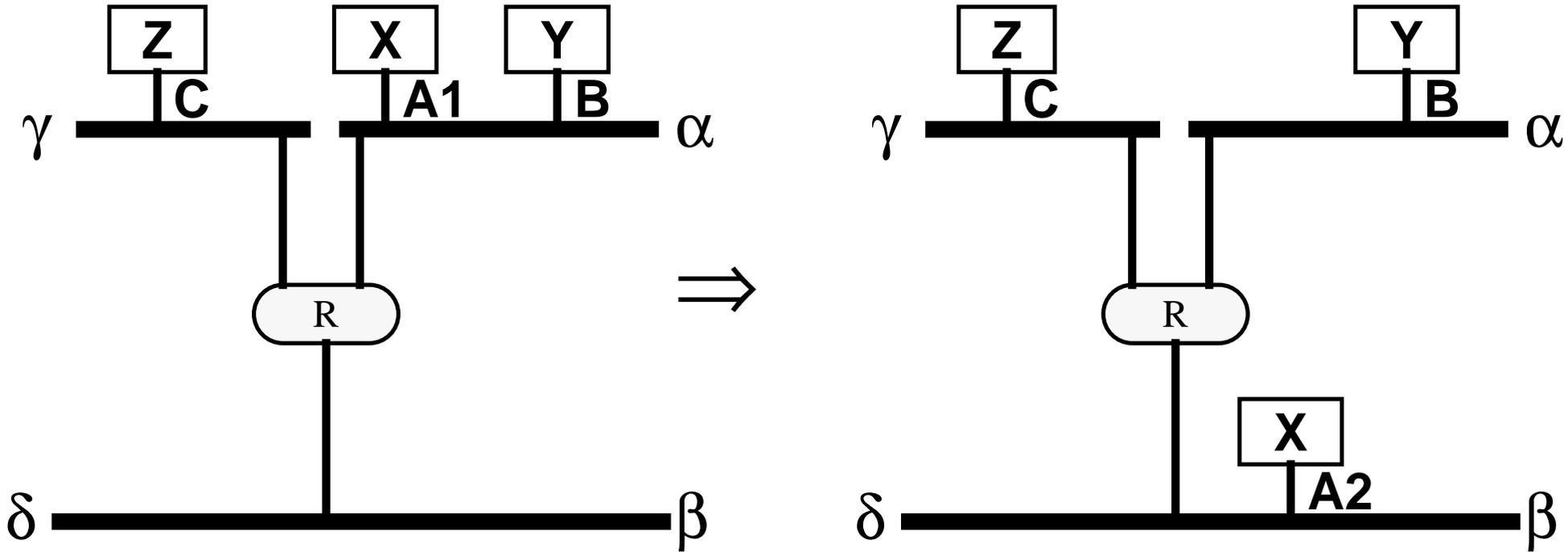
Figure 97. How do we know it is the same X?

When X moves to its new location (A2)

- Why should it get service?
- How do we know it is the same X? (Or even that it is X?)

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 622 of 663
Internetworking/Internetteknik

Establishing Identity

When a node arrives on a network it must identify itself

- mechanism: typically via a challenge response protocol

- Who should it identify itself to? Answer: The MSR ≡ Mobility Support Router
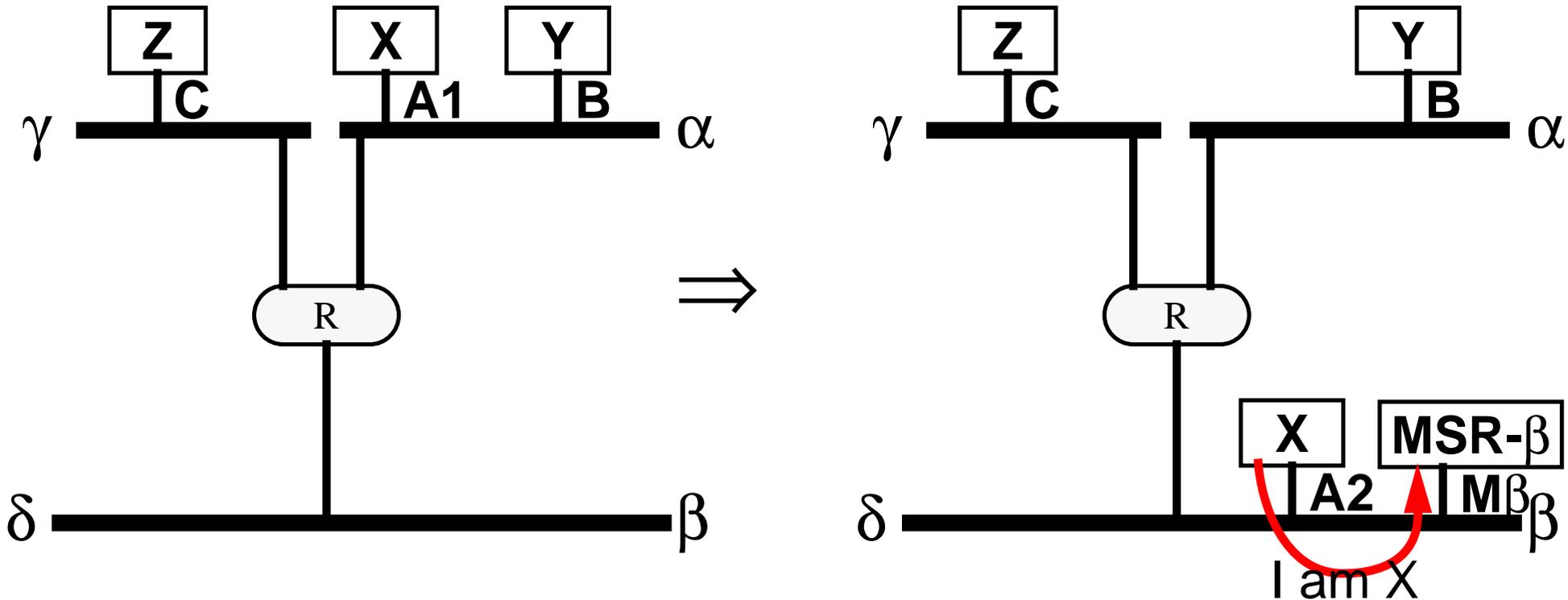
Figure 98. How do we know it is the same X?

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 623 of 663
Internetworking/Internetteknik

How did it know to send the "I am" message to the MSR?

- **When a node arrives on a network it listens for broadcasts from MSRs**
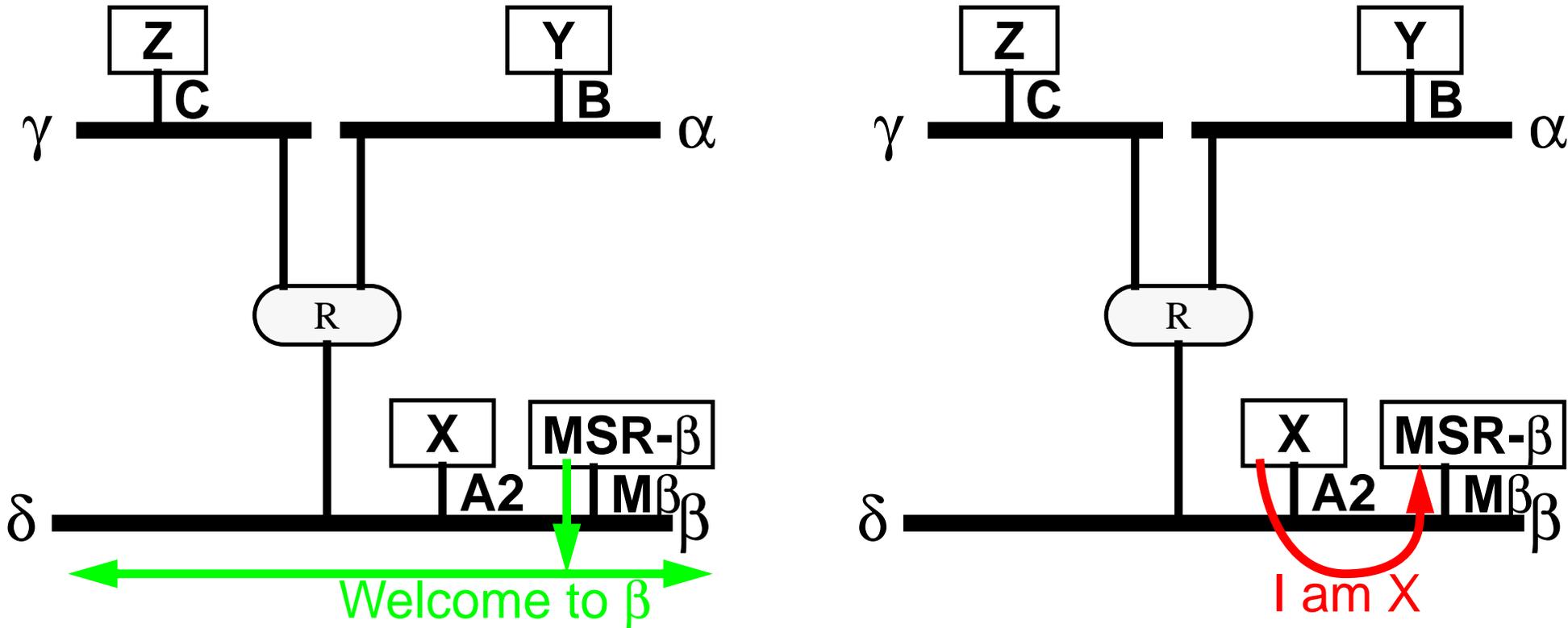
Figure 99. "Welcome (Greeting)" messages answered by "I am" messages

These broadcast "Welcome" messages advertise:

- the presence of an MSR (and its MAC address)
- advertise one or more networks it provides connectivity to

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 624 of 663
Internetworking/Internetteknik

Could the MSR functionality be collocated with the router?

- **When a node arrives on a network it listens for broadcasts from MSRs**
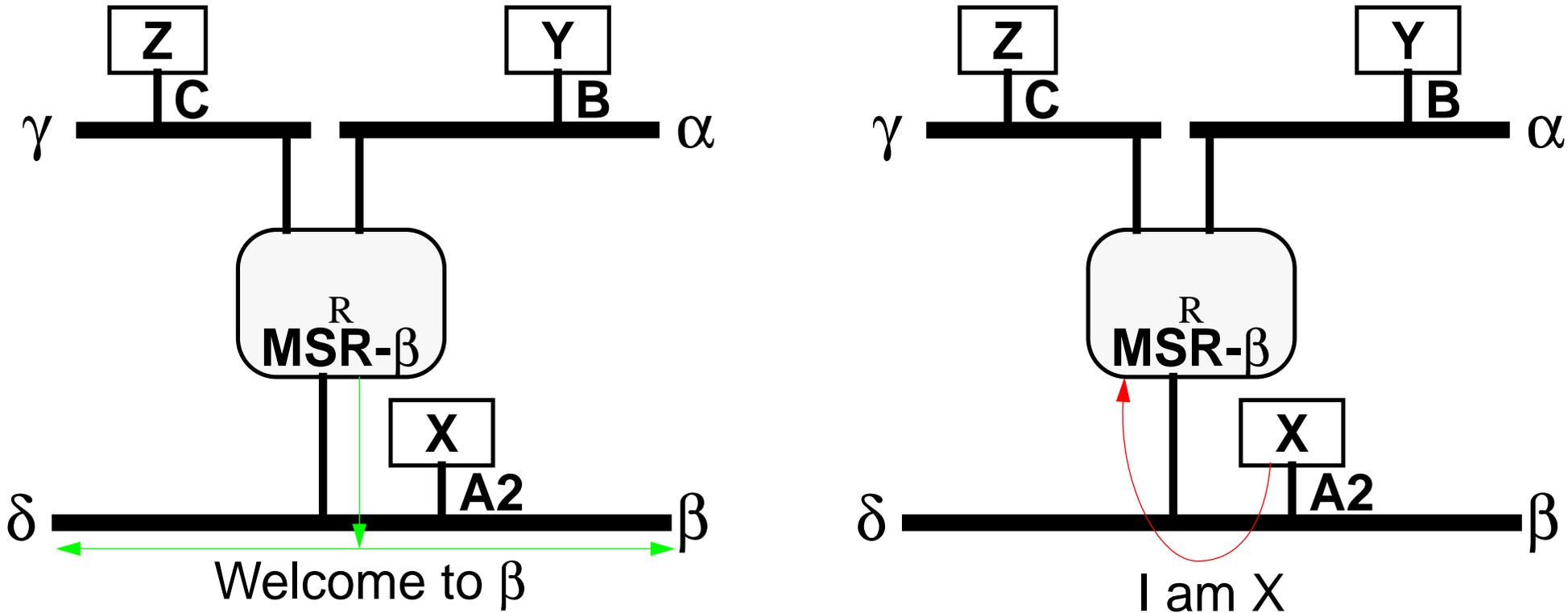
Figure 100. "Welcome (Greeting)" messages from router answered by "I am" messages

    ✘ Requires updating **all** of the routers on network segments which are going to support mobility to be updated.

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 625 of 663
Internetworking/Internetteknik

Getting Service

Once it's identity is know, the policy **question must be ask: Should X get service?**

The policy question and its answer may involve:

- roaming agreements (generally reciprocal agreements),

- current traffic loads,

- anticipated traffic loads,

- mobile user's priority/class/… ,

- … .

The question of authentication, authorization, and accounting (AAA) for mobile users is addressed in [108].

See also IEEE 802.1x Port Based Network Access Control

http://www.ieee802.org/1/pages/802.1x.html

Back to the original problem: Z wants to send a message to X

Initially X is located at A1 then it moves to A2.
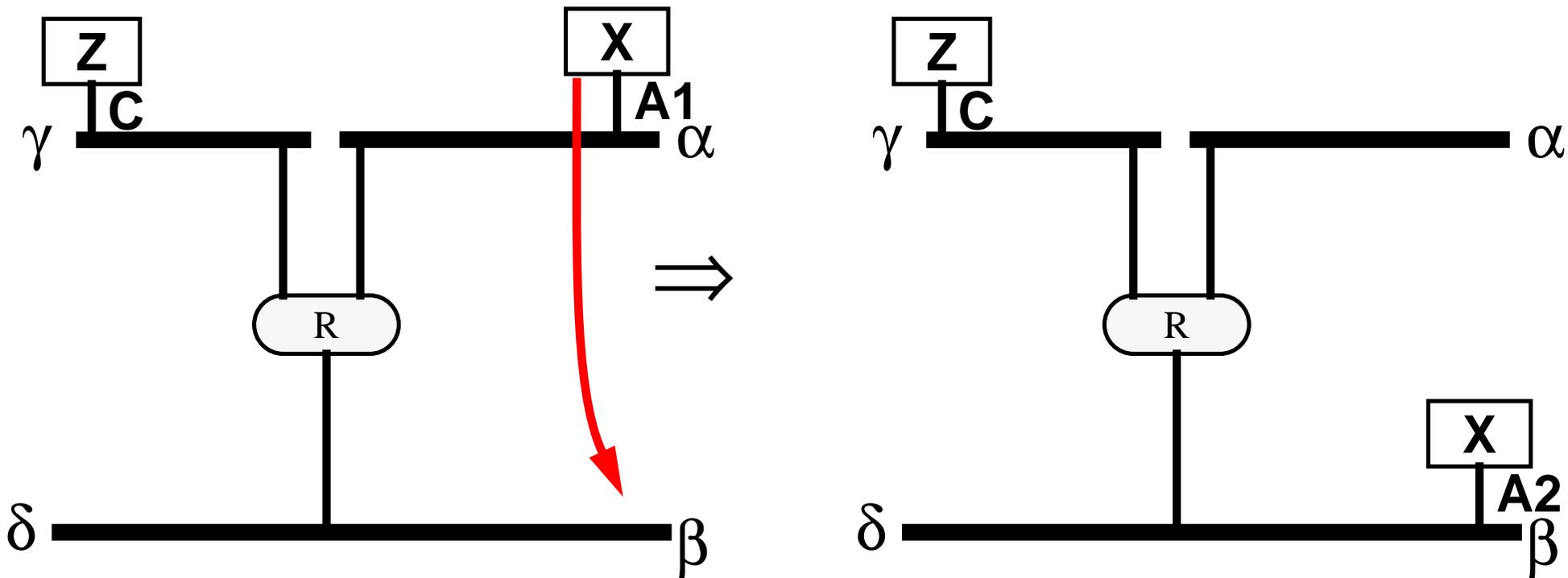
Figure 101. X moves from A1 to A2, Z not aware of Mobility

There are several alternatives.

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 627 of 663
Internetworking/Internetteknik

Alternative 1

Initially X is located at A1 then it moves to A2.
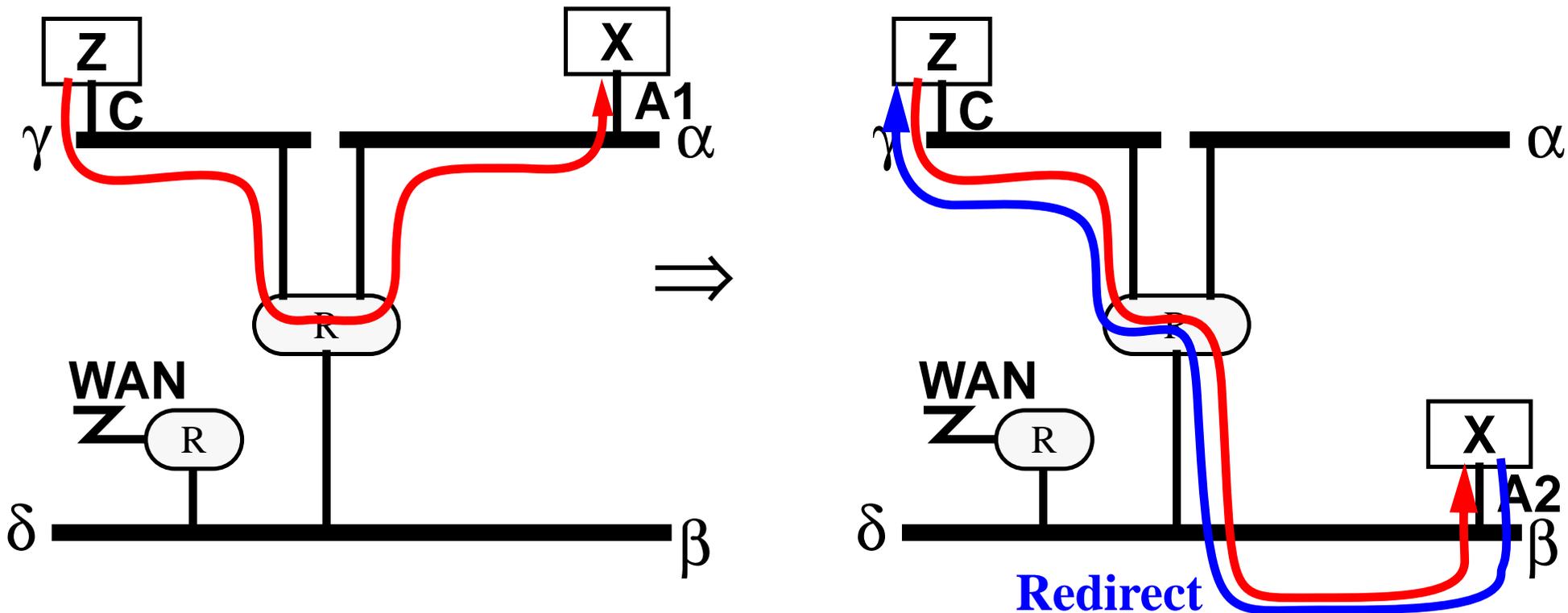
Figure 102. X must send a redirect message to Z, to tell it it's new address A2.

✘ Z must be aware of where X currently is.

✘ X must get a new local address A2 (How? perhaps DHCP)

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 628 of 663
Internetworking/Internetteknik

Alternative 2

Initially X is located at A1 then it moves to A2.
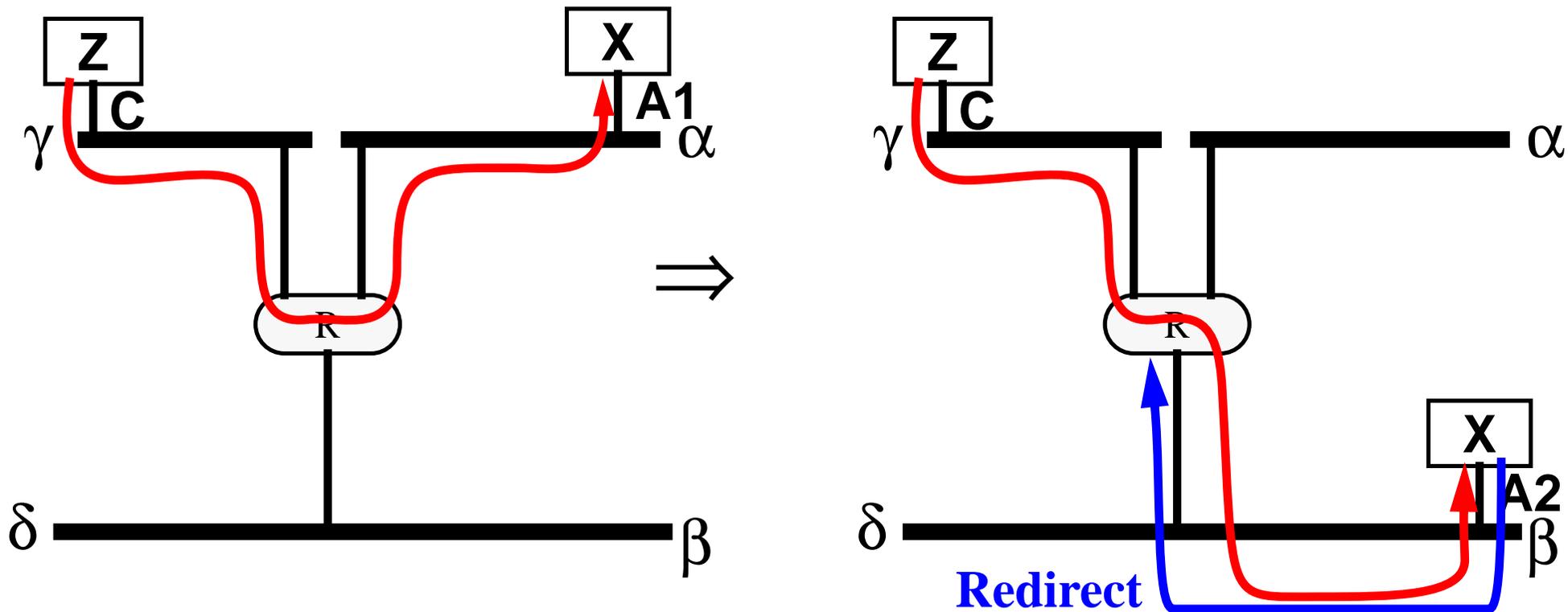
Figure 103. X must send a redirect message to the **Router**, to tell it it's new address A2 (rather than A1).

- ✘ Router must now perform host specific routing.
- ✘ X must get a new local address A2 (How? perhaps DHCP)

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 629 of 663
Internetworking/Internetteknik

Alternative 3

Initially X is located at A1 then it moves to A2.
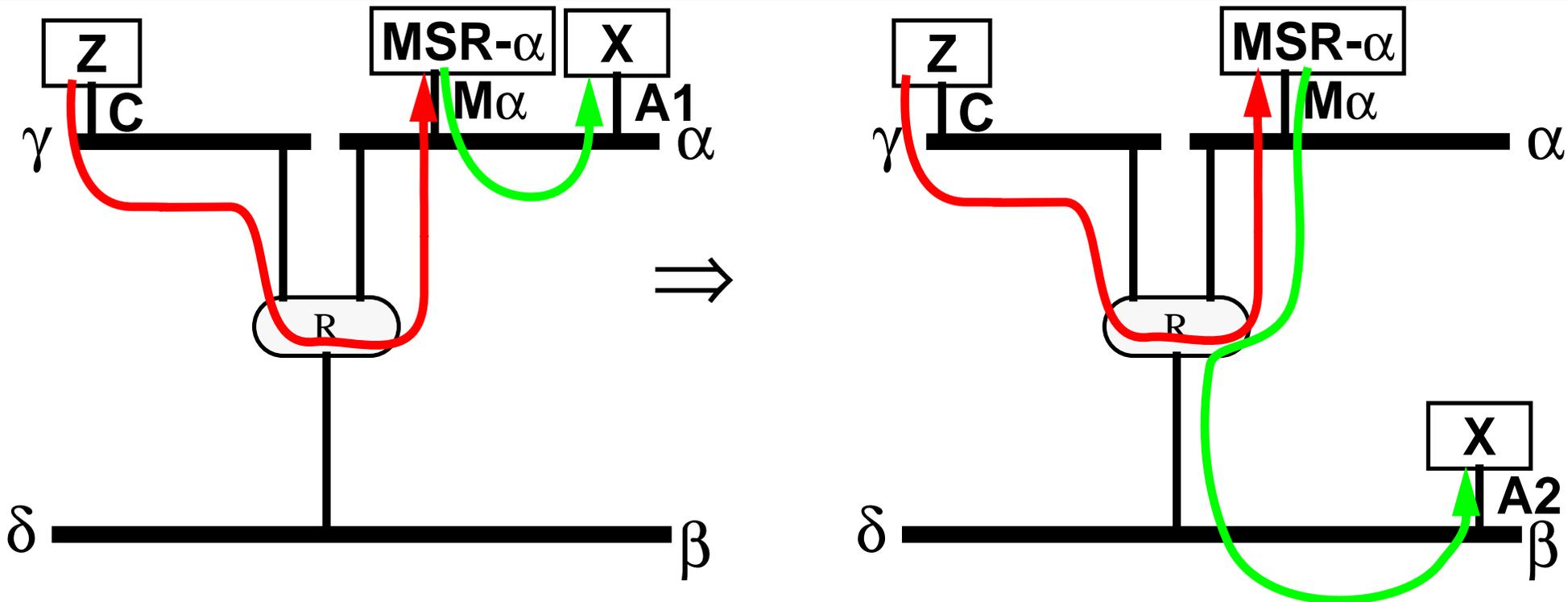
Figure 104. X must send a redirect message to a Mobility Support Router (**MSR-α**), to tell it it's new address A2 (rather than A1).

✘ **MSR-α** must now perform host specific routing.

✘ X must get a new local address A2 (How? perhaps DHCP)

✔ Z is now completely unaware of the move.

✔ Router R is now completely unaware of the move (except for twice the traffic over the link to/from α).

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 630 of 663
Internetworking/Internetteknik

Alternative 4

Initially X is located at A1 then it moves to A2.
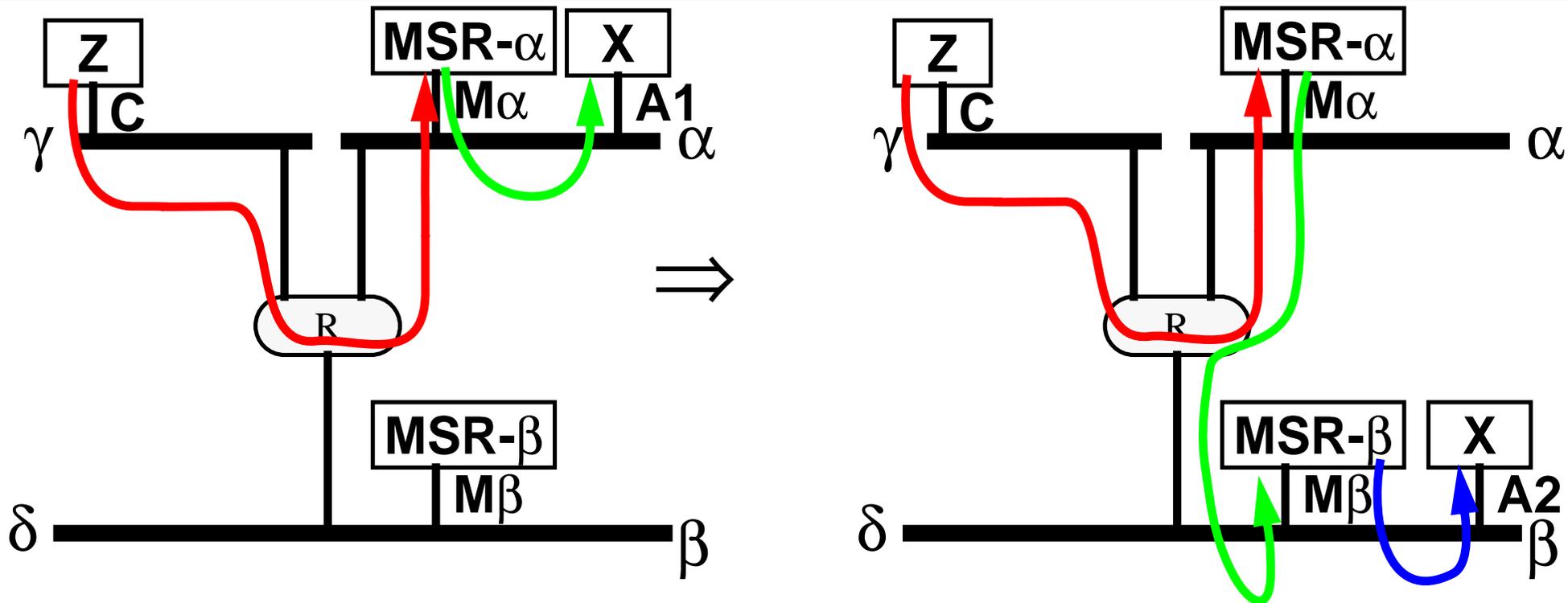
Figure 105. X sends a message to **MSR-β**, to get it's new address A2 and says it's old MSR was **MSR-α**.

- ✘ **MSR-α** must now perform host specific routing to **MSR-β** (which can provide the local address A2)
- ✔ Z is now completely unaware of the move - it always sends traffic to **MSR-α**.
- ✔ If X moves again, Z does not change where it sends traffic to & traffic need not go via **MSR-β** - it will go directly from **MSR-α** to the MSR responsible for the new segment.

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 631 of 663
Internetworking/Internetteknik

Initially X is located at A1 then it moves to A2 and then moves to A3.
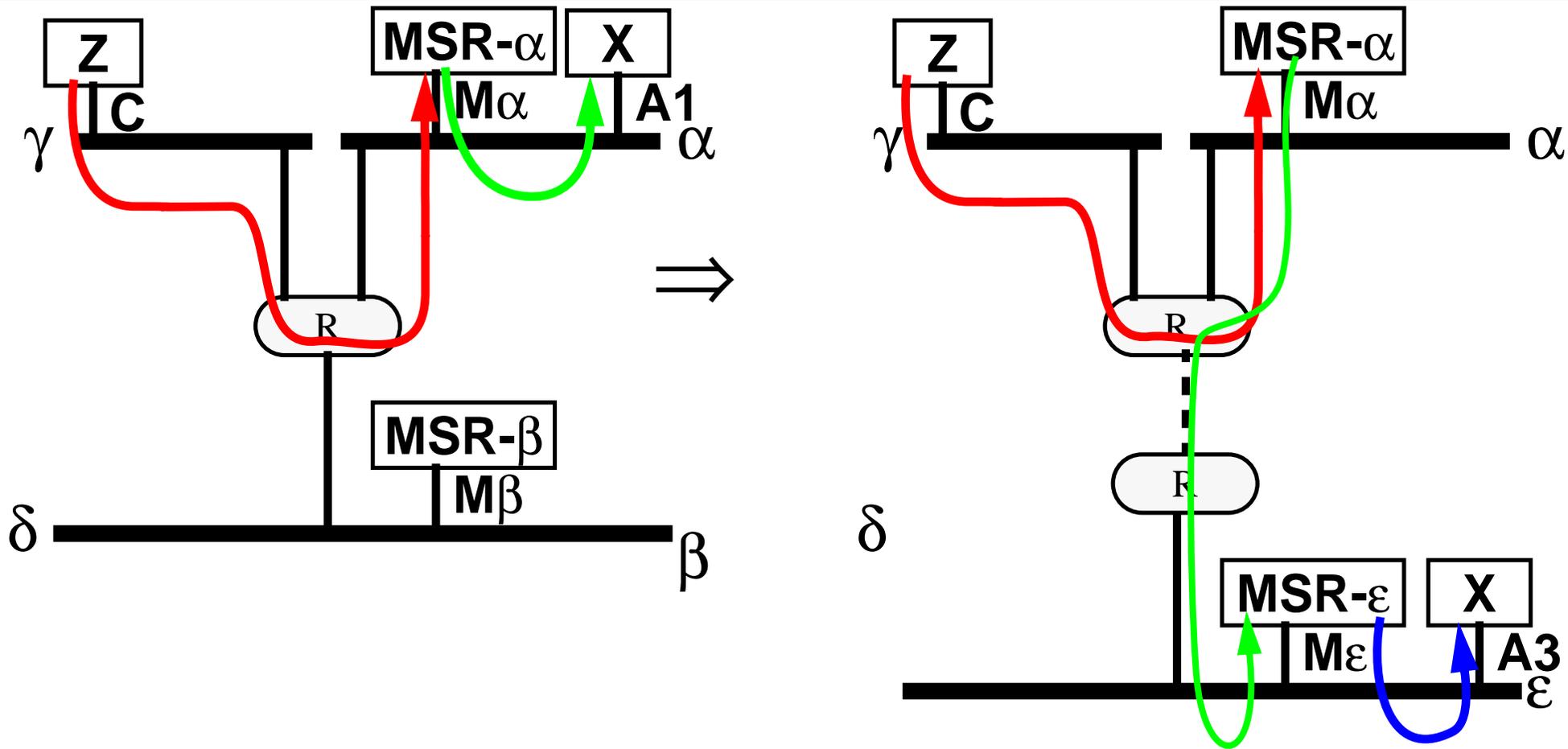
Figure 106. X sends a message to **MSR-ε**, to get it's new address A3 and says it's old MSR was **MSR-α**.

✔ The traffic from **MSR-α** to **MSR-β** or **MSR-α** to **MSR-ε** can be encapsulated, using for example IP in IP (written IP-IP) encapsulation. Thus none of the intervening routers needs know about mobility.

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 632 of 663
Internetworking/Internetteknik

How does Z know to send things to MSR-α?

It does **not** know to do this! ⇒ Z simply sends the packet to the network address of X.

But what is the (real) network address of X?.
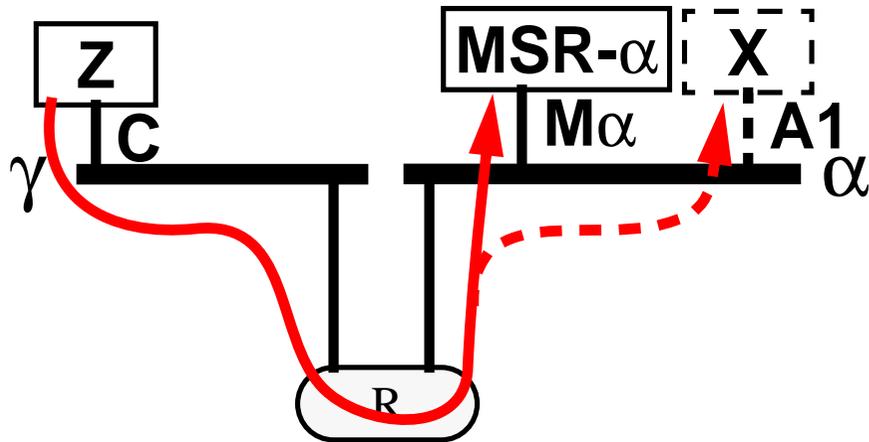
- X's address is A1

- A1 is an address on network α

- MSA-α intercepts packets addressed to A1 and forwards them if X is not currently present on the network α
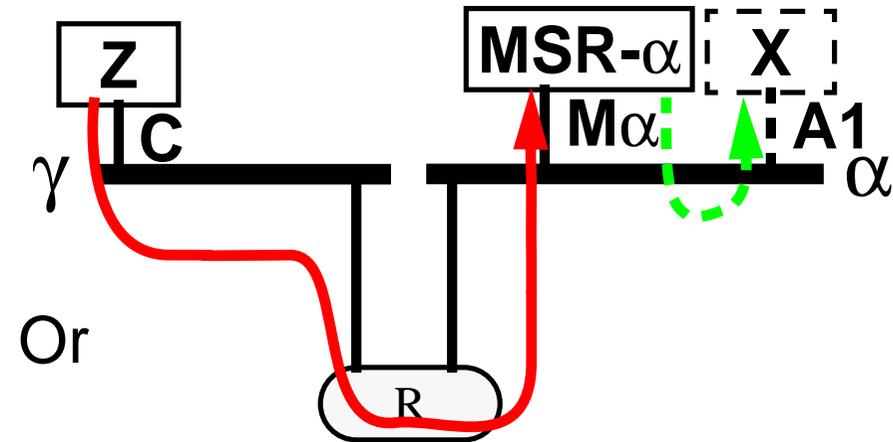
- X's address is {Mobile-Network,X}

- A1 is a temporary address on network α

- MSA-α routes {Mobile-Network,X} packets to **A1** when X is **local** and to another **MSR** when it is **non-local**

Figure 107. X's address - either on either an actual network or on a virtual network

✔ In the first case ("actual" network addresses), either the hosts and routers have to be changed, or MSRs are necessary to intercept and reroute the packets.

✔ In the virtual network case, we use the MSRs to implement mobility for nodes on a virtual mobile network.

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 633 of 663
Internetworking/Internetteknik

What happens in the case of wireless networks?
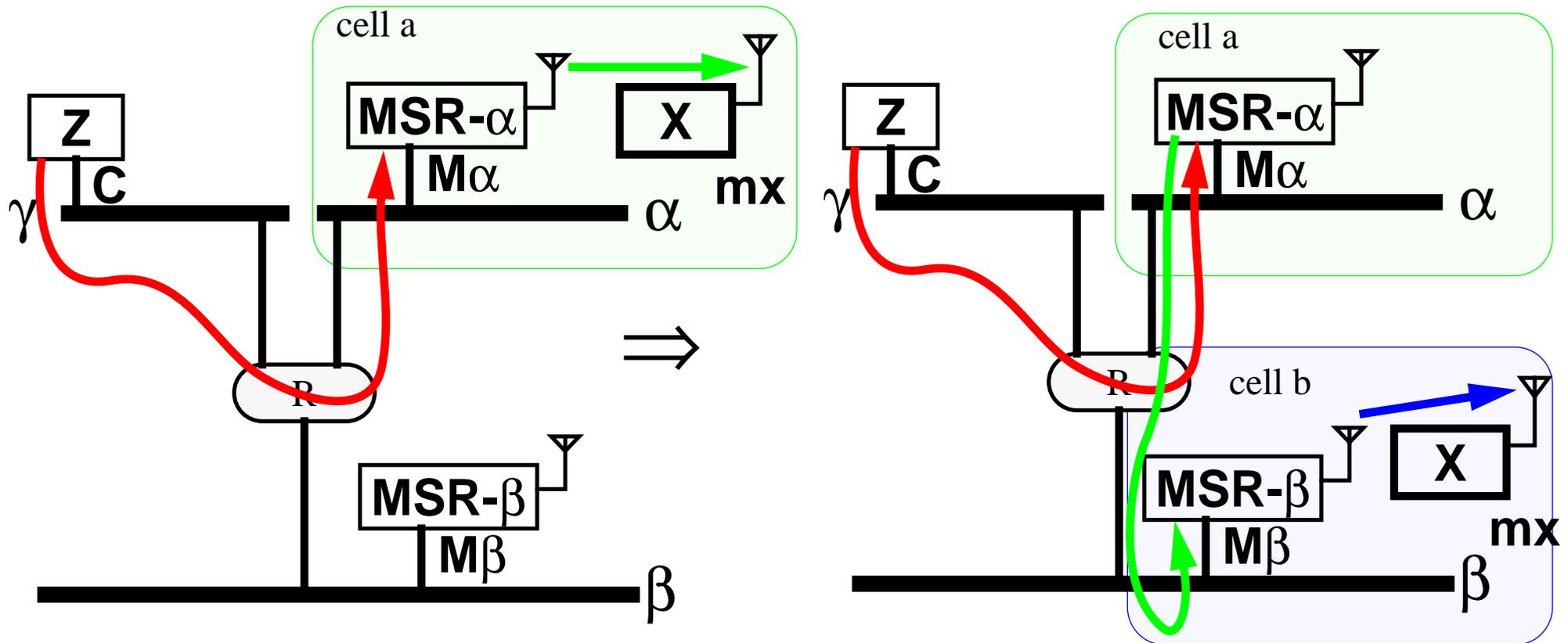
Figure 108. X moves from the cell a to the cell b

- The wireless cells are implemented by a basestation co-located with the MSR.
- Note that X retains it's mobile network address "mx".

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 634 of 663
Internetworking/Internetteknik

Wireless Local Area Networks
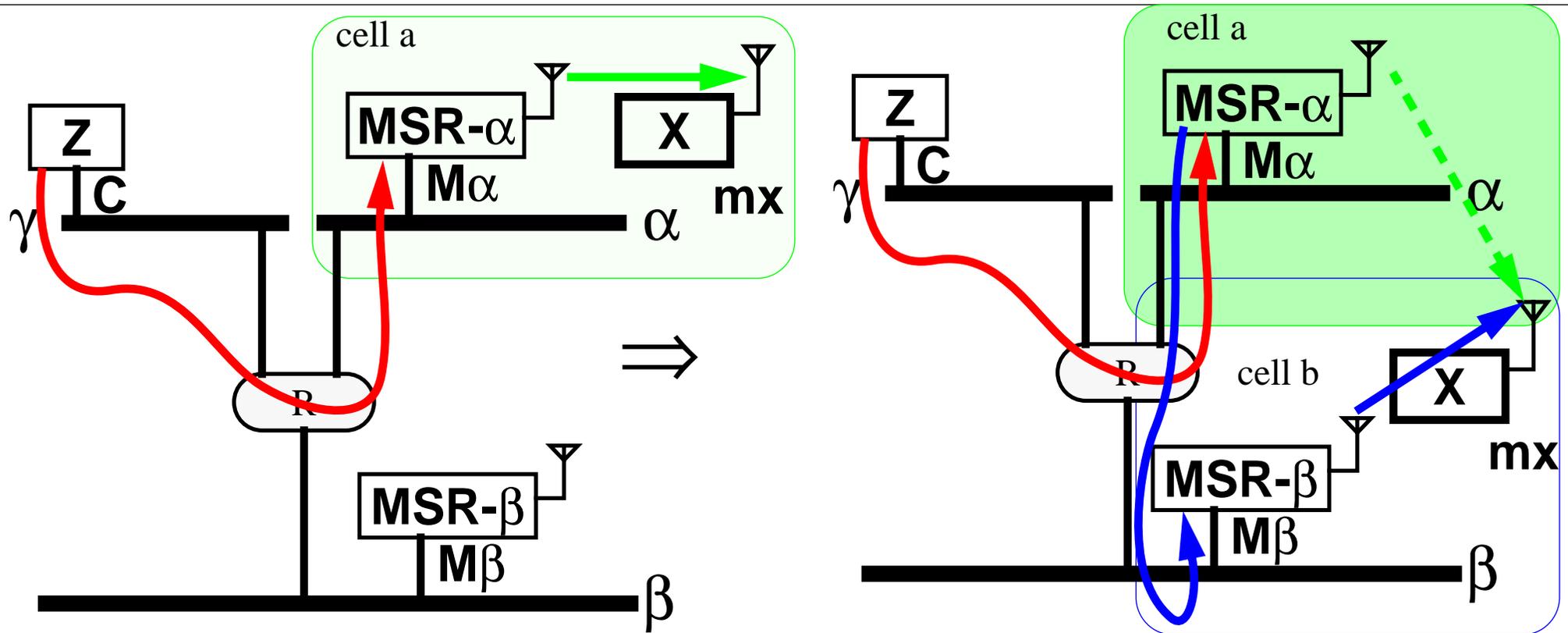
Figure 109. X moves from the cell a to the cell b, but is still reachable by cell a!

- Mobile network address "mx" is reachable from both MSR-α and MSR-β.
- This could not occur in the wired case (unless there were multiple interfaces), since X would have to disconnect from network α to connect to network β.
- If the cell size is small the movement between cells could be frequent (and caused by other events, such as a new user, a door moving, …).

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 635 of 663
Internetworking/Internetteknik

Wireless WANs
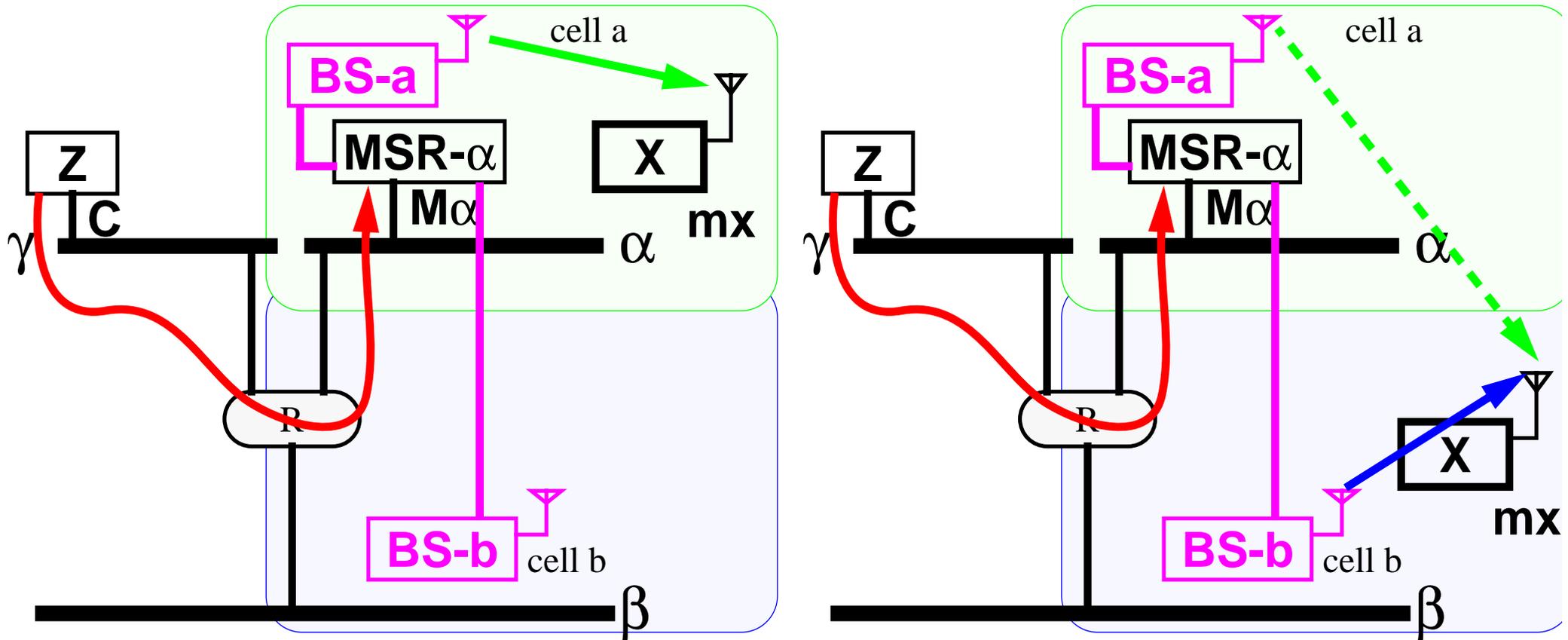
Figure 110. X moves from the cell a to the cell b, but may still reachable by cell a - but both cells are part of the same network

- Basestation-a, basestation-b, … are all part of the same network and it is up to this network to select which cell a mobile is in and which basestation will be used to communicate with it.

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 636 of 663
Internetworking/Internetteknik

Simulcasting in Wireless Local Area Networks (WLANs)
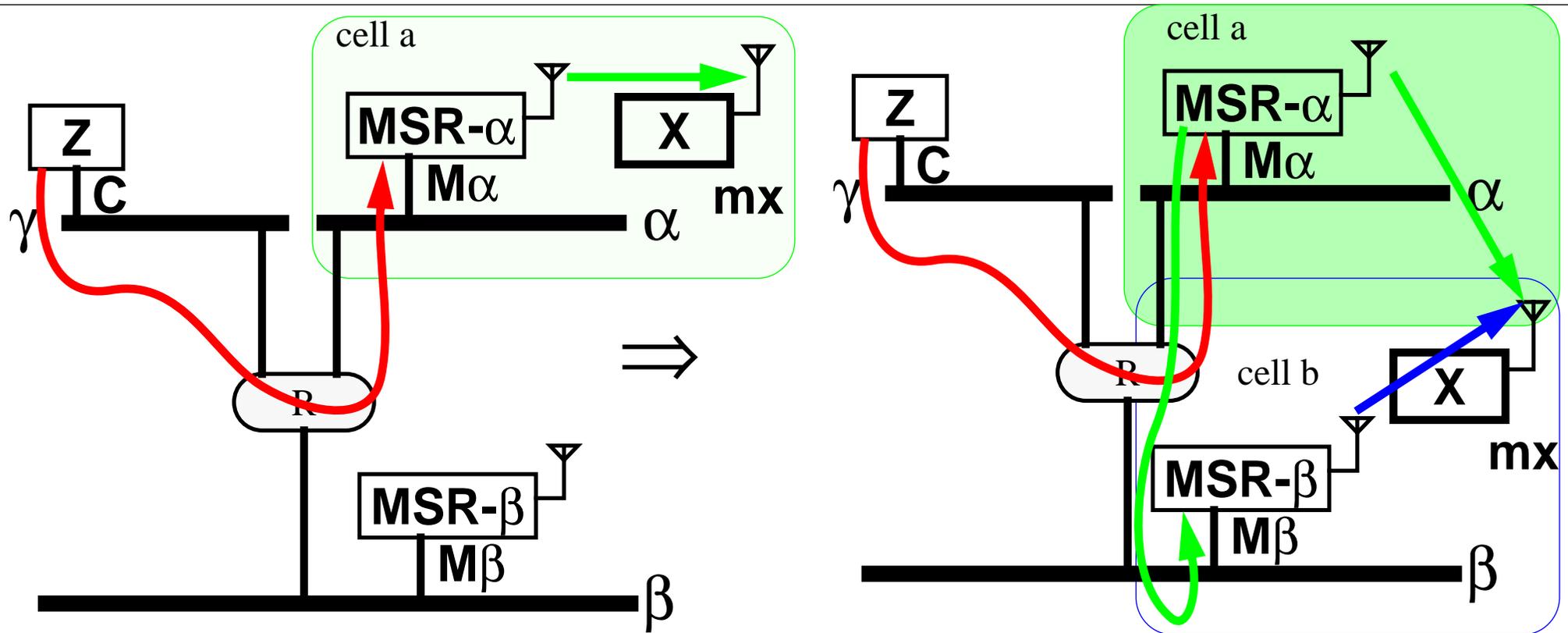
Figure 111. X is moving from the cell a to the cell b

- Mobile network address "mx" is partially reachable from both MSR-α and MSR-β - thus we will send packets via both MSR-α and MSR-β. This insures:

  ✔ Lower probability of packet loss (important if we must provide low latency and high reliability - such as is needed for voice and some other services)

  ✘ increases traffic in both cells

Maguire
maguire@it.kth.se

Objectives of Mobile IP
2006.03.13

Mobile IP 637 of 663
Internetworking/Internetteknik

Mobile IP Standardization Effort

- Originally proposed by Columbia University, IBM, etc.
- Internet Engineering Task Force (IETF) Mobile-IP working group
  - *http://www.ietf.org/html.charters/mobileip-charter.html*

Mobile-IP standard status:

- RFCs:
  - Mobile-IPv4 (RFC 2002) IP Mobility Support; RFC 2003: IP Encapsulation within IP; RFC 2004, RFC 2005, RFC 2006, etc.
  - Mobile-IPv6
- Many Drafts related to v4 & v6:
  - Mobile IP NAI Extension, AAA Registration Keys for MIP, Registration Keys for Route Optimization, Mobile IP Challenge/Response Extensions, CDMA2000 Extension to MIP, Cellular IP, Regional Tunnel Management, Hierarchical MIP Handoffs, etc.
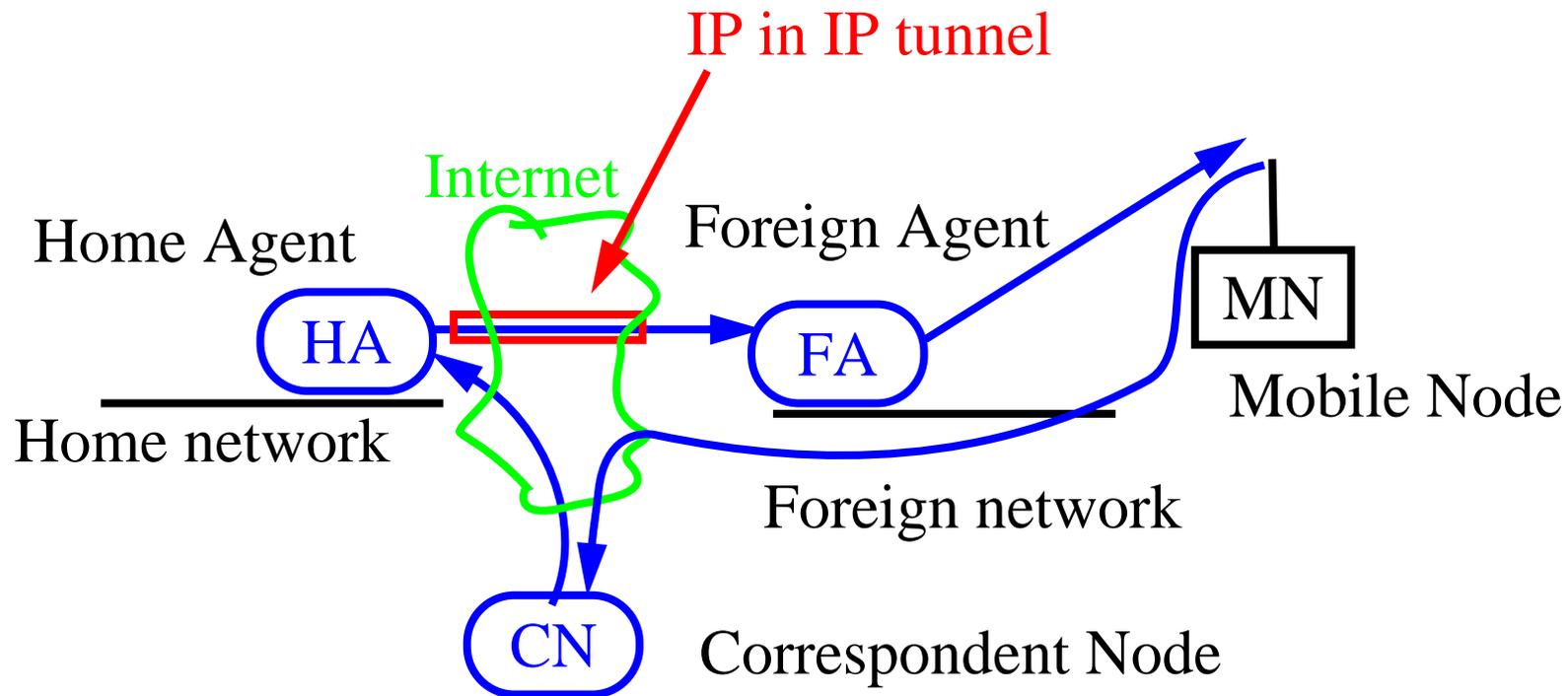
Maguire
maguire@it.kth.se

Mobile IP Standardization Effort
2006.03.13

Mobile IP 638 of 663
Internetworking/Internetteknik

A Mobile-IP(V4) Scenario

CN sends packet to MN's home network (because that is where its IP address is logically located), HA intercepts them and forwards them inside an IP-in-IP tunnel to the Care of Address (CoA) where the FA forwards them to the MN.

Traffic from the MN can go directly to the CN (**unless** there is ingress filtering) ⇒ triangle routing

Maguire
maguire@it.kth.se

A Mobile-IP(V4) Scenario
2006.03.13

Mobile IP 639 of 663
Internetworking/Internetteknik

A Mobile-IP(V6) Scenario

CN sends packet to MN's home network (because that is where its IP address is logically located), HA intercepts them and forwards them inside an IP-in-IP tunnel to the Care of Address (CoA) which is the MN's address in the foreign network.

However, the MN can tell the CNabout its **current** address via a binding update (BU), now traffic can flow both ways directly between the CN and MN.

Maguire
maguire@it.kth.se

A Mobile-IP(V6) Scenario
2006.03.13

Mobile IP 640 of 663
Internetworking/Internetteknik

IP-in-IP Encapsulation

In-in-IP vs. Minimal encapsulation - the major difference is the first puts the whole IP packet inside another, while the later tries to only put a minimal header inside along with the original data portion of the IP packet.

For details see

- IP Encapsulation within IP, RFC 2003 [106]
- Minimal Encapsulation within, IP RFC 2004 [107]

Maguire
maguire@it.kth.se

IP-in-IP Encapsulation
2006.03.13

Mobile IP 641 of 663
Internetworking/Internetteknik

Tunneling IP Datagrams

Both home agents and foreign agents (v4) must support tunneling datagrams using IP-in-IP encapsulation and decapsulation.

MNs that use a co-located COA must also support decapsulation (v6).

Maguire
maguire@it.kth.se

Tunneling IP Datagrams
2006.03.13

Mobile IP 642 of 663
Internetworking/Internetteknik
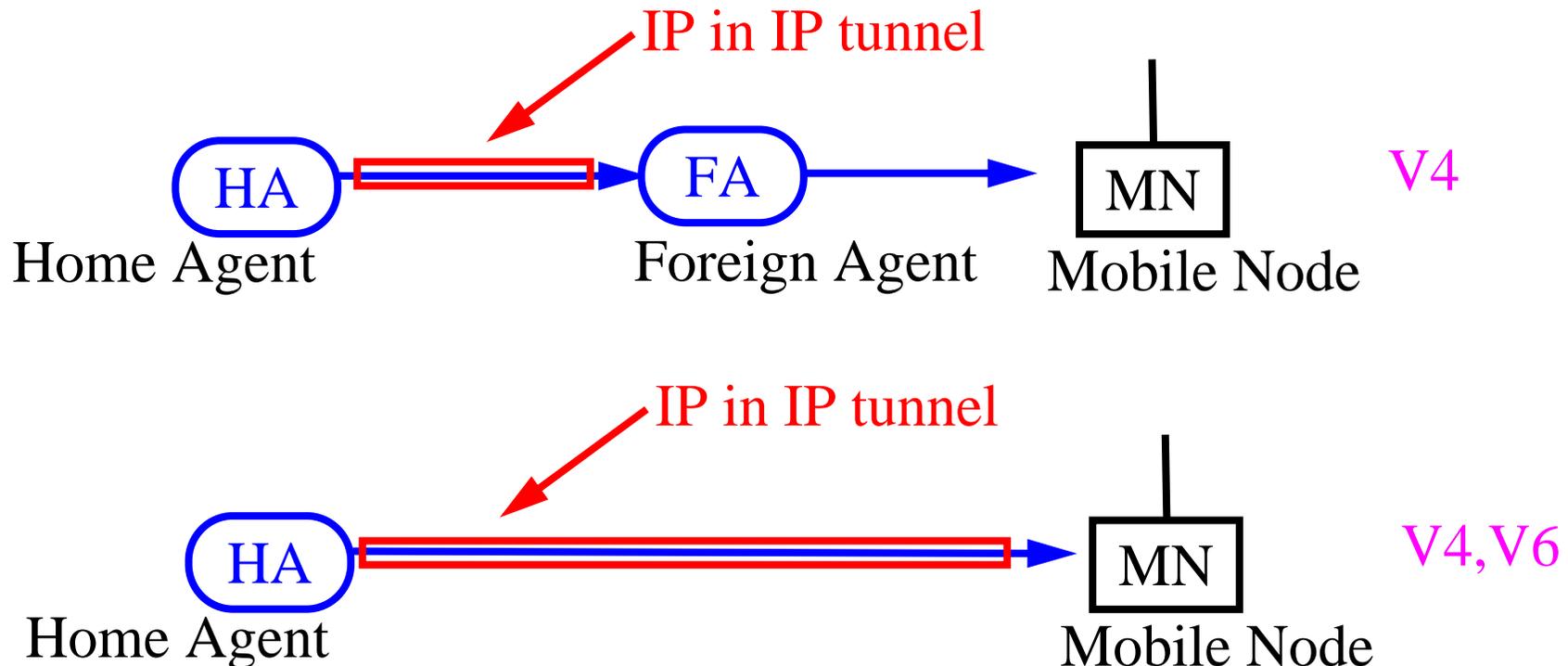
Temporary Address Assignment

Two types of temporary Care-Of-Address:

- ## Foreign agent care-of address (V4)
    - a care-of address provided by a foreign agent through its Agent Advertisement messages.
- ## Co-located care-of address (V4, V6)
    - a care-of address acquired by the mobile node as a local IP address through some external means, eg. dynamically acquired as a temporary address through dynamic host configuration protocol (DHCP) RFC 1541, or the address may be owned by the MN as a long-term address for its use while visiting this foreign network.

Maguire
maguire@it.kth.se

Temporary Address Assignment
2006.03.13

Mobile IP 643 of 663
Internetworking/Internetteknik

Agent Discovery

Why Agent Discovery?

Methods an MN can use to determine whether it is currently at its home network or a foreign network. By:

- ## Agent Advertisement
  - periodic transmissions (beacons) sent by a mobility agent (rate limited to max. 1/s).
- ## Agent Solicitation
  - Send by an MN to discover agents.

Maguire
maguire@it.kth.se

Agent Discovery
2006.03.13

Mobile IP 644 of 663
Internetworking/Internetteknik
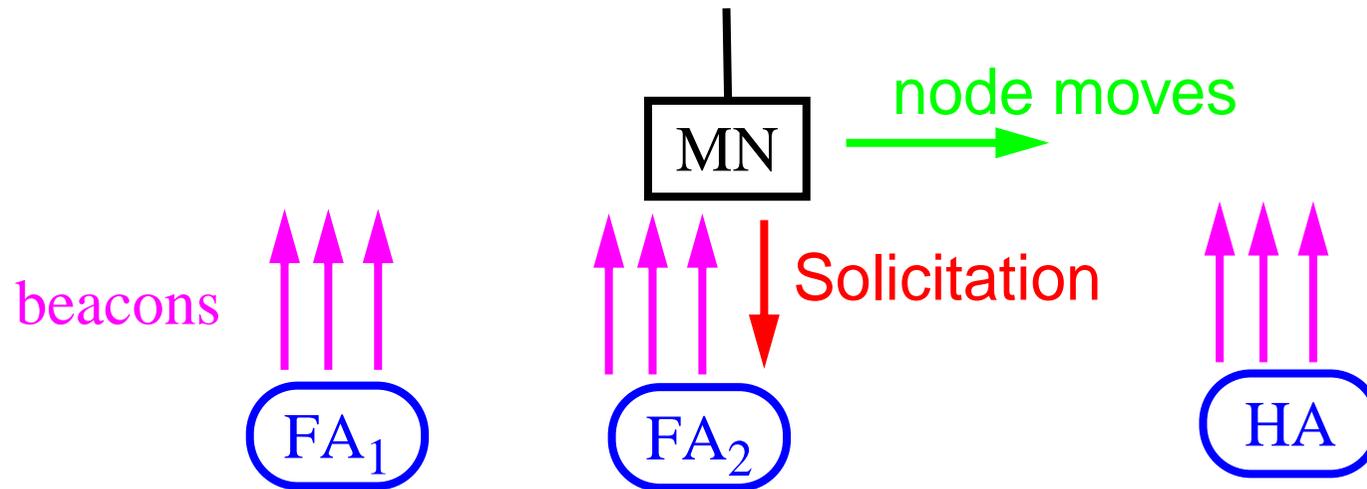
Agent Advertisement Message Format

Extension of an ICMP router advertisement

| 0 | | 8 | | 16 | | 24 | | 31 |

TYPE (16)	Length	Sequence Number
Lifetime	CODE	Reserved
Care of Address* {the number is determined by the length field; must be at least 1 of F bit set}		

Bit	Name	Meaning
0	R	Registration with this foreign agent (or another foreign agent on this link) is required; using a co-located care-of address is not permitted.
1	B	Busy. Foreign agent not accepting registrations from additional mobile nodes.
2	H	Agent offers service as a home agent.
3	F	Agent offers service as a foreign agent.
4	M	Agent implements receiving tunneled datagrams that use minimal encapsulation
5	G	Agent implements receiving tunneled datagrams that use GRE encapsulation
6	V	Agent supports Van Jacobson header compression over the link with any registered mobile node.
7		reserved (must be zero)

Maguire
maguire@it.kth.se

Agent Advertisement Message Format
2006.03.13

Mobile IP 645 of 663
Internetworking/Internetteknik

Registration Message Format

0	8	16	24	31

TYPE (1 or 3)	FLAGS	Lifetime
Home Address		
Home Agent		
Care of Address* {the number is determined by the length field; must be at least 1 of F bit set}		
Identification		
Extensions		

Bit	Name	Meaning
0	S	Simultaneous bindings, this is an additional address for the mobile
1	B	Broadcast datagrams. Home agent to tunnel any broadcast packets it receives to the mobile.
2	D	Mobile using co-located care-of address and will decapsulation itself
3	M	Mobile requests home agent to use Minimal encapsulation.
4	G	Mobile requests home agent to use GRE encapsulation.
5	V	Mobile node requests that agent use Van Jacobson header compression.
6-7		reserved (must be zero)

Maguire
maguire@it.kth.se

Registration Message Format
2006.03.13

Mobile IP 646 of 663
Internetworking/Internetteknik

MN Requirements

An MN must have:

- home address, netmask,
- mobility security association for each HA.

For each pending registration, MN maintains the following information:

- link-layer address of the FA to which the Registration Request was sent
- IP destination address of the Registration Request
- Care-of address used in the registration
- remaining lifetime of the registration

Maguire
maguire@it.kth.se

MN Requirements
2006.03.13

Mobile IP 647 of 663
Internetworking/Internetteknik

FA Requirements (v4)

- Each FA must be configured with a **care-of-address**.

- Must maintain a **visitor list** with following information:
  - Link-layer source address of the mobile node
  - IP Source Address (the MN's Home Address)
  - UDP Source Port
  - Home Agent address
  - Requested registration Lifetime
  - Identification field

This visitor list acts much like a Visitor Location Register (VLR) in a cellular system.

Maguire
maguire@it.kth.se

FA Requirements (v4)
2006.03.13

Mobile IP 648 of 663
Internetworking/Internetteknik

HA Requirements

Each HA must have:

- the home address and mobility security association of each authorized MN that it is serving as a home agent.

Must create or modify its **mobility binding list** entry containing:

- Mobile node's CoA (or CoAs in the case of simultaneous bindings)
- Identification field from the Registration Request
- Remaining Lifetime of the registration

The mobility binding list acts much like a Home Location Register (HLR) in a cellular system.

Maguire
maguire@it.kth.se

HA Requirements
2006.03.13

Mobile IP 649 of 663
Internetworking/Internetteknik

Optimization Problem
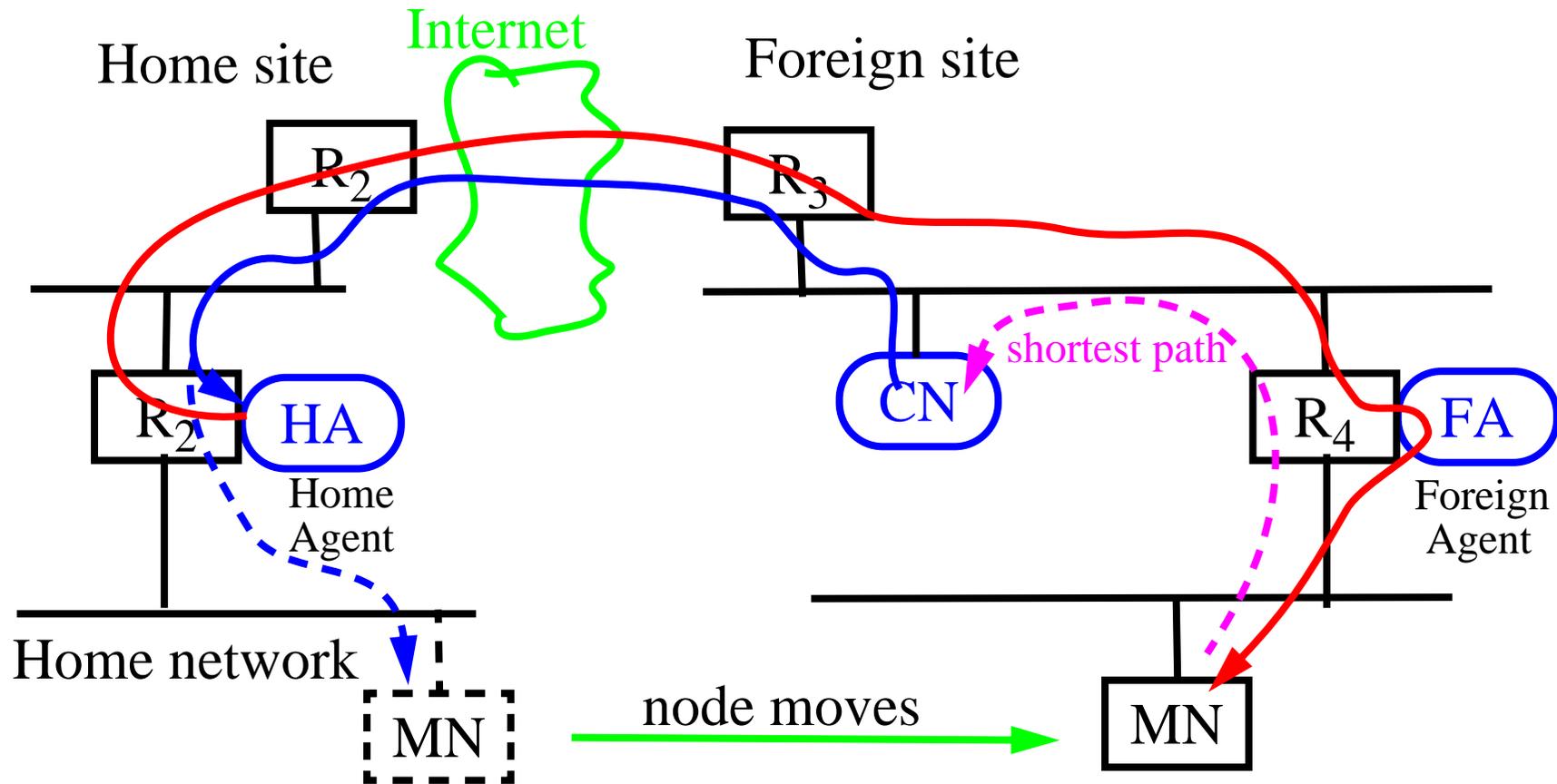
We can **not** follow the shortest path in Mobile IPv4 because the CN will always send it via our home network. However, we may be able to use the shortest path from the MN to the CN.

Maguire
maguire@it.kth.se

Optimization Problem
2006.03.13

Mobile IP 650 of 663
Internetworking/Internetteknik

Problems of Mobile IP (RFC2002)

- Only provides basic "macro mobility" support
- Not developed for cellular systems   } ⟹ Cellular
- No interface defined between cellular systems
    - (e.g. between Mobile-IP/HLR/VLR)   } Micro Mobility
- No handover support

- Weak in security   } Security
- No key distribution mechanism

- Route optimization problems   ⟹ Optimization

- No QoS, real-time support, (DiffServ, RSVP)   ⟹ QoS and Real-time

- …

Maguire
maguire@it.kth.se

Problems of Mobile IP (RFC2002)
2006.03.13

Mobile IP 651 of 663
Internetworking/Internetteknik

Mobile IP Problems and Development

- Cellular Micro Mobility:
  - CDMA2000 Extension to MIP
  - Cellular IP
  - Regional Tunnel Management
  - Hierarchical MIPv6 Handoffs
  - MIP based Micro Mobility Mgt

- Security:
  - Mobile IP NAI Extension
  - AAA Registration Keys for MIP
  - Registration Keys for Route   Optimization
  - Mobile IP Challenge/Response Extensions

- Route Optimization:
  - Route optimization for MIPv4, v6

- Real-time QoS:
  - No solution yet

Maguire
maguire@it.kth.se

Mobile IP Problems and Development
2006.03.13

Mobile IP 652 of 663
Internetworking/Internetteknik

CDMA2000 Extension to Mobile IP

A draft entitled: Mobile IP Based Micro Mobility Management Protocol in the Third Generation Wireless Network, by 3Com, Alcatel, Cisco, Ericsson, Lucent, Nortel, Motorola, Samsung, etc.

RFC2002
Mobile IP

Internet

HA

PDSN

FA

GRE Encapsulation and reverse tunneling during Registration

RNN

MN

Radio Network Node

CN

Maguire
maguire@it.kth.se

CDMA2000 Extension to Mobile IP
2006.03.13

Mobile IP 653 of 663
Internetworking/Internetteknik

Wireless IP Network Architecture

Maguire
maguire@it.kth.se

Wireless IP Network Architecture
2006.03.13

Mobile IP 654 of 663
Internetworking/Internetteknik
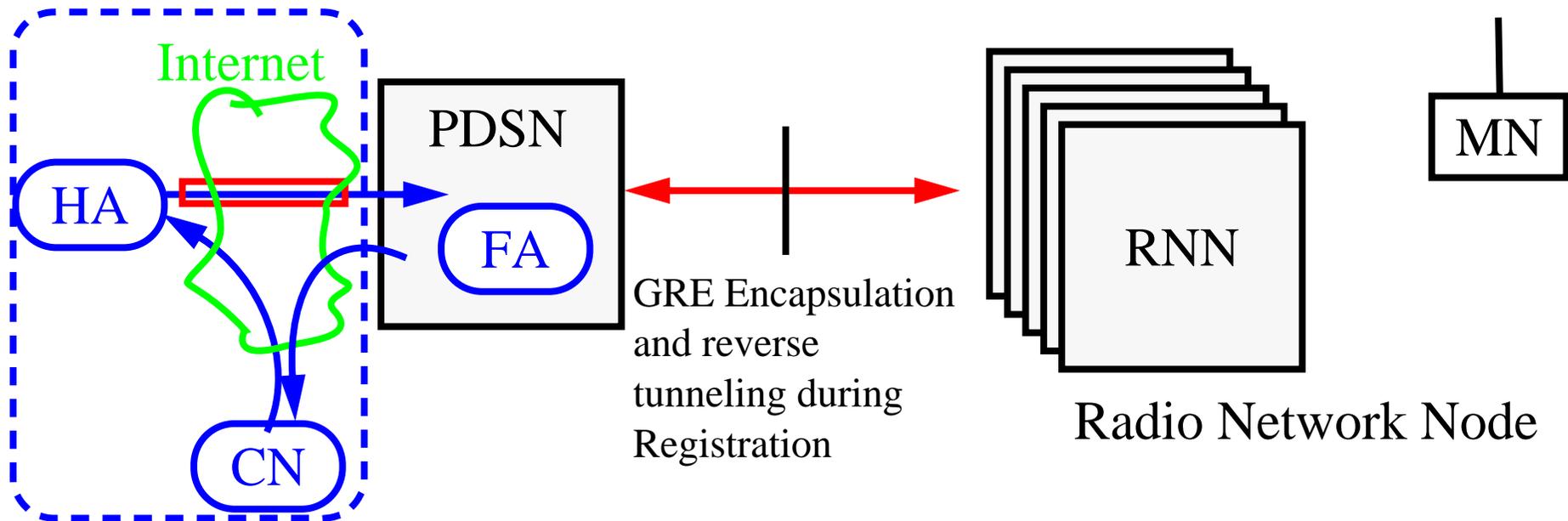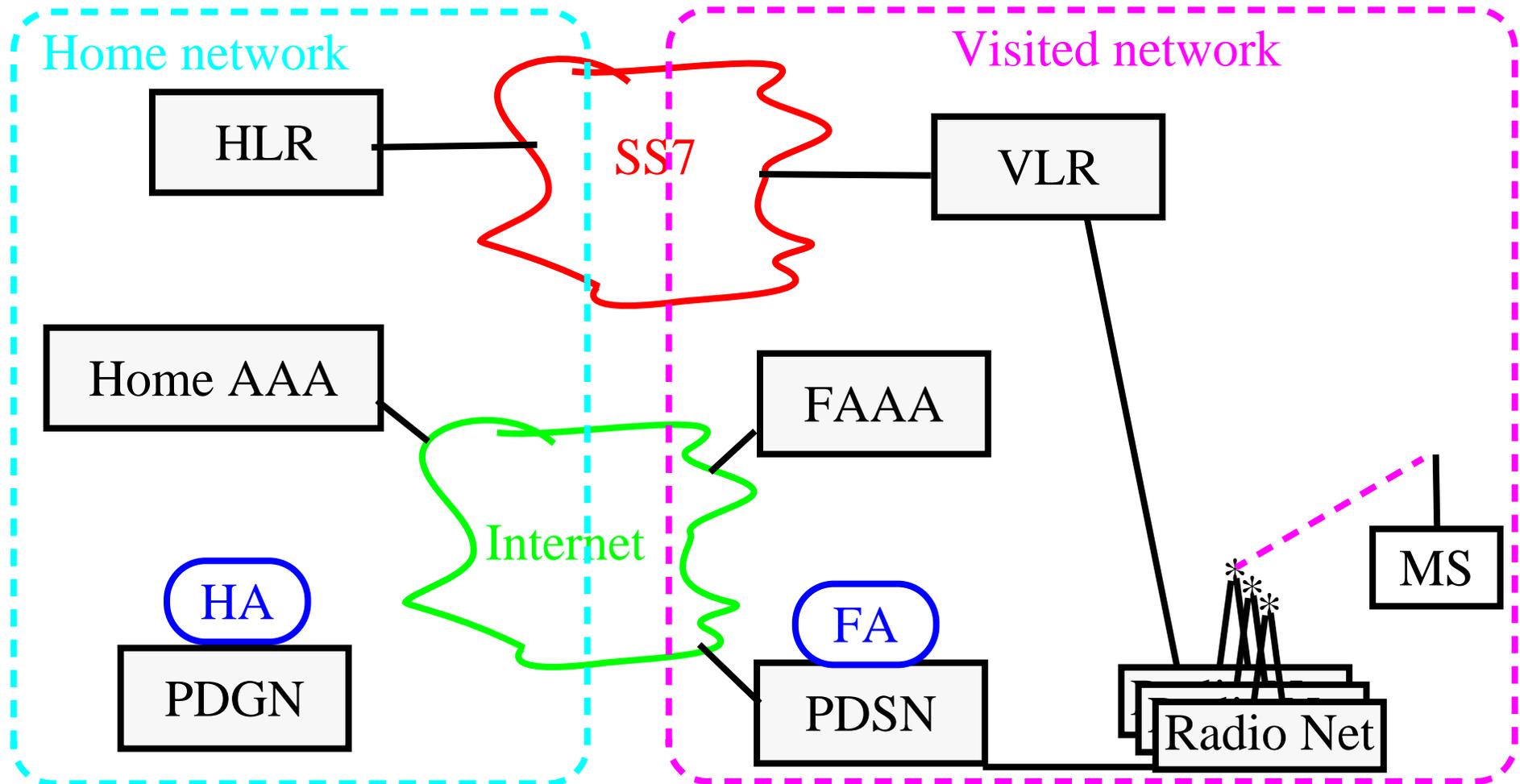
Cellular IP (CIP)

HAWAII extension is similar to Cellular IP.

Maguire
maguire@it.kth.se

Cellular IP (CIP)
2006.03.13

Mobile IP 655 of 663
Internetworking/Internetteknik
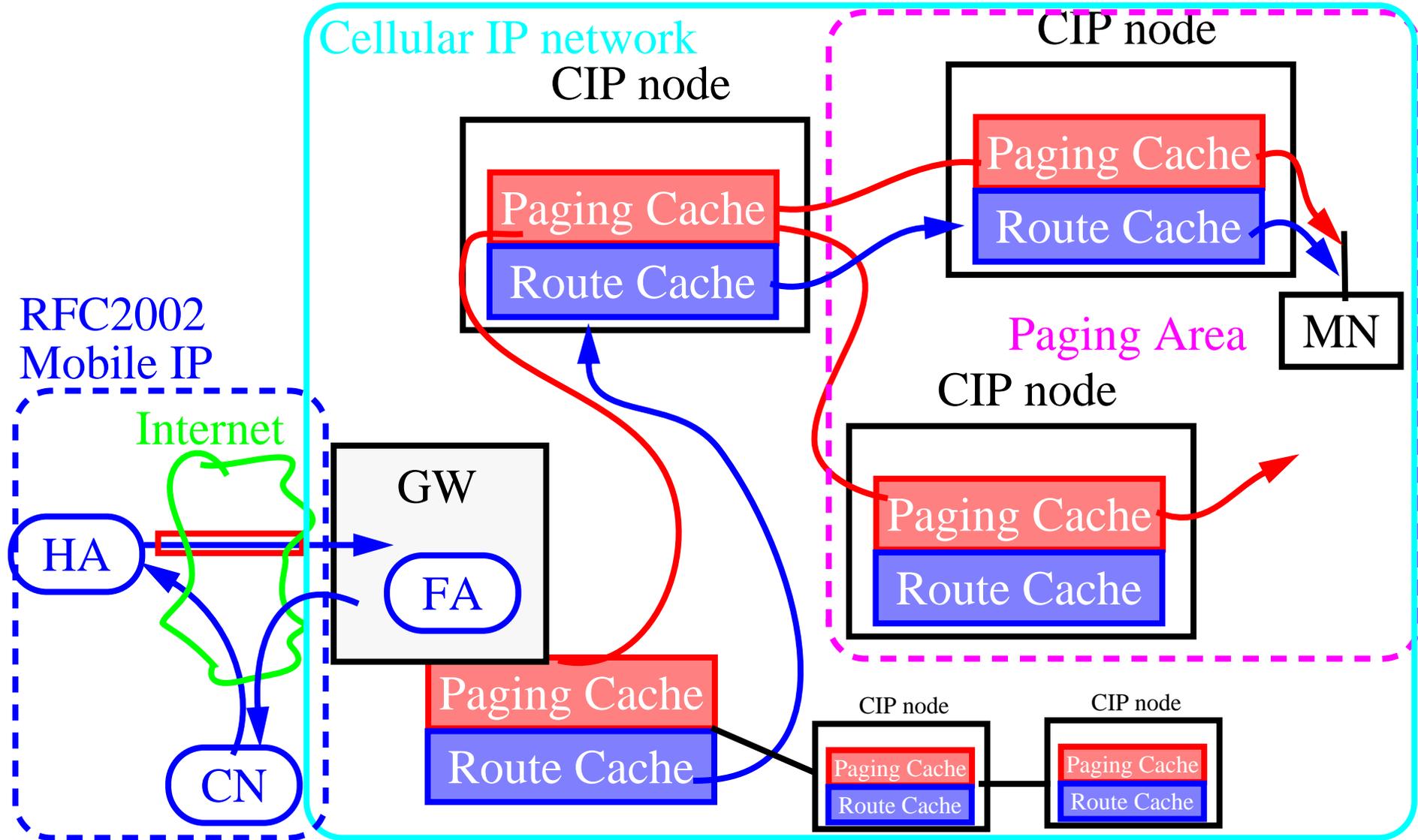
Cellular IP (CIP): Handover

HAWAII extension is similar to Cellular IP.

Maguire
maguire@it.kth.se

Cellular IP (CIP): Handover
2006.03.13

Mobile IP 656 of 663
Internetworking/Internetteknik
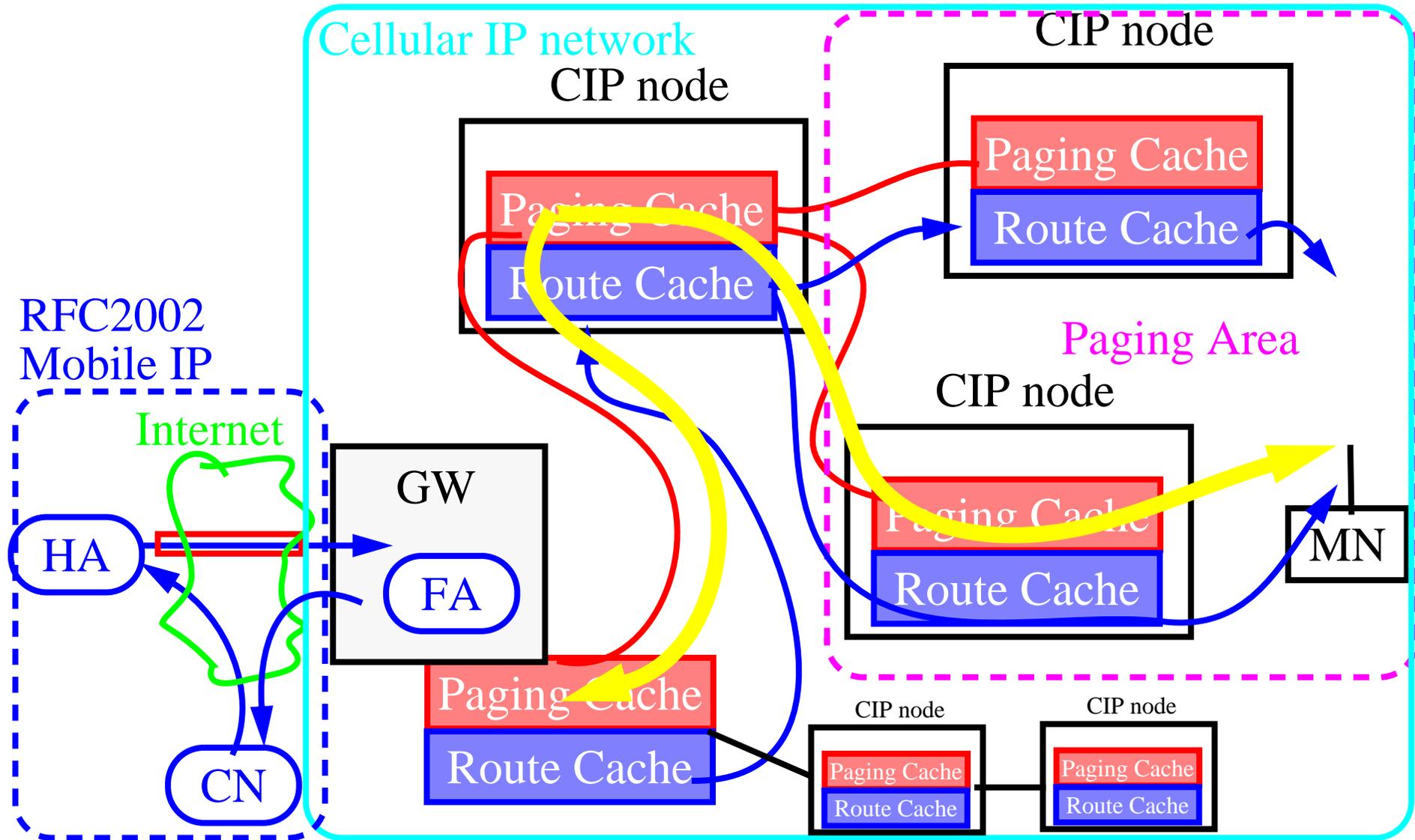
Cellular IP (CIP): Location Update

HAWAII extension is similar to Cellular IP.
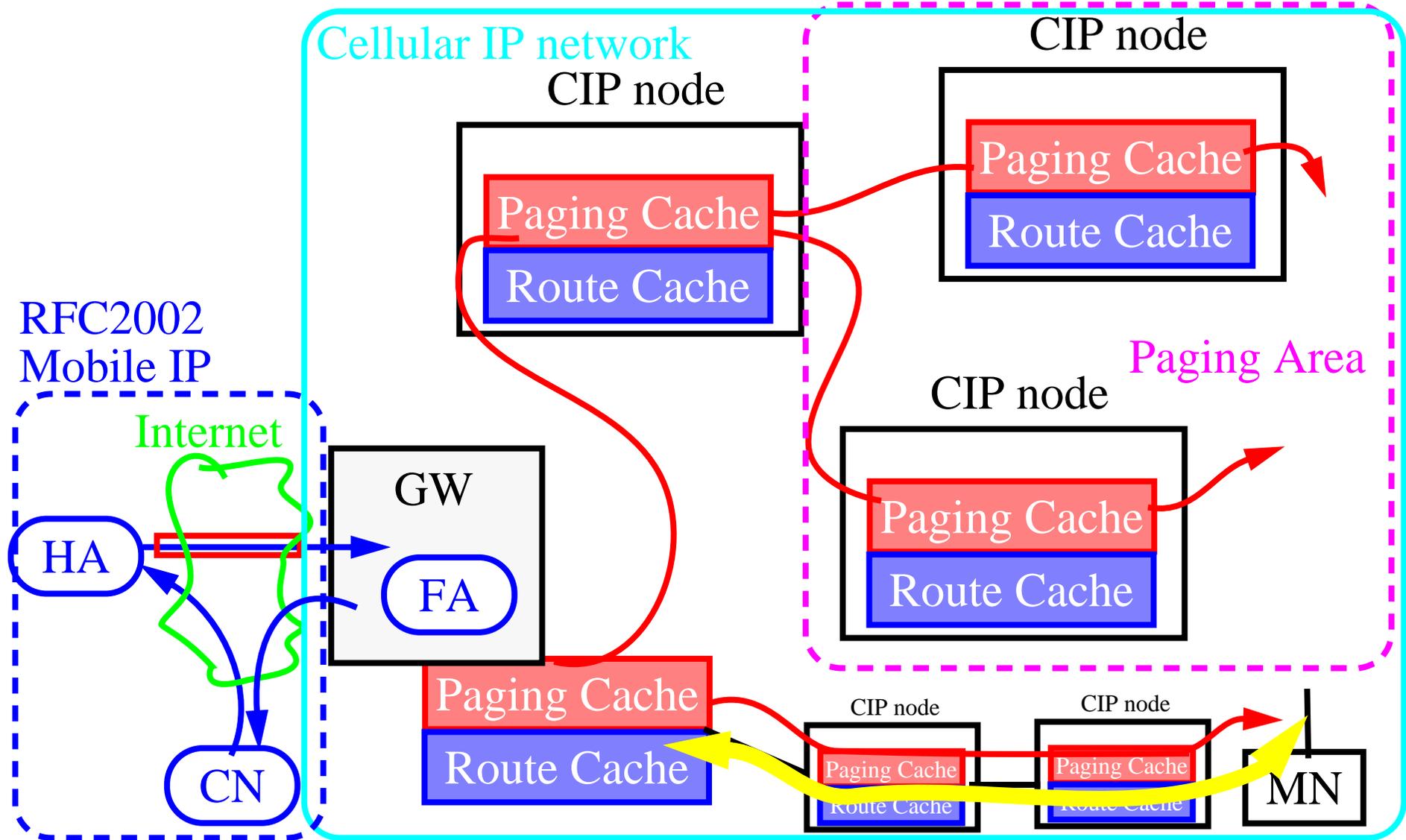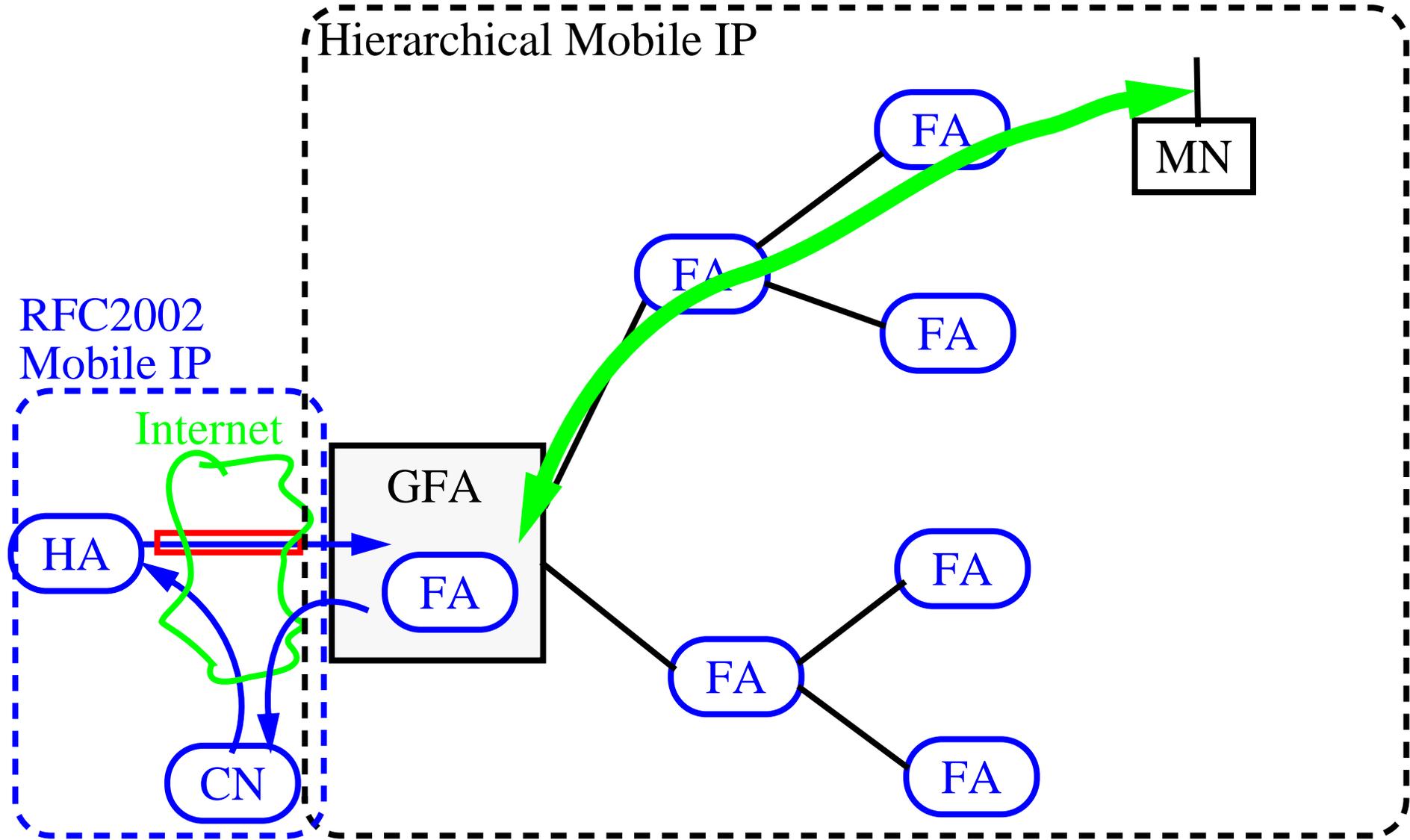
Maguire
maguire@it.kth.se

Cellular IP (CIP): Location Update
2006.03.13

Mobile IP 657 of 663
Internetworking/Internetteknik

Hierarchical FA and Regional Tunneling

Maguire
maguire@it.kth.se

Hierarchical FA and Regional Tunneling
2006.03.13

Mobile IP 658 of 663
Internetworking/Internetteknik

Hierarchical FA and Regional Tunneling

Maguire
maguire@it.kth.se

Hierarchical FA and Regional Tunneling
2006.03.13

Mobile IP 659 of 663
Internetworking/Internetteknik

Hierarchical FA and Regional Tunneling

Maguire
maguire@it.kth.se

Hierarchical FA and Regional Tunneling
2006.03.13

Mobile IP 660 of 663
Internetworking/Internetteknik
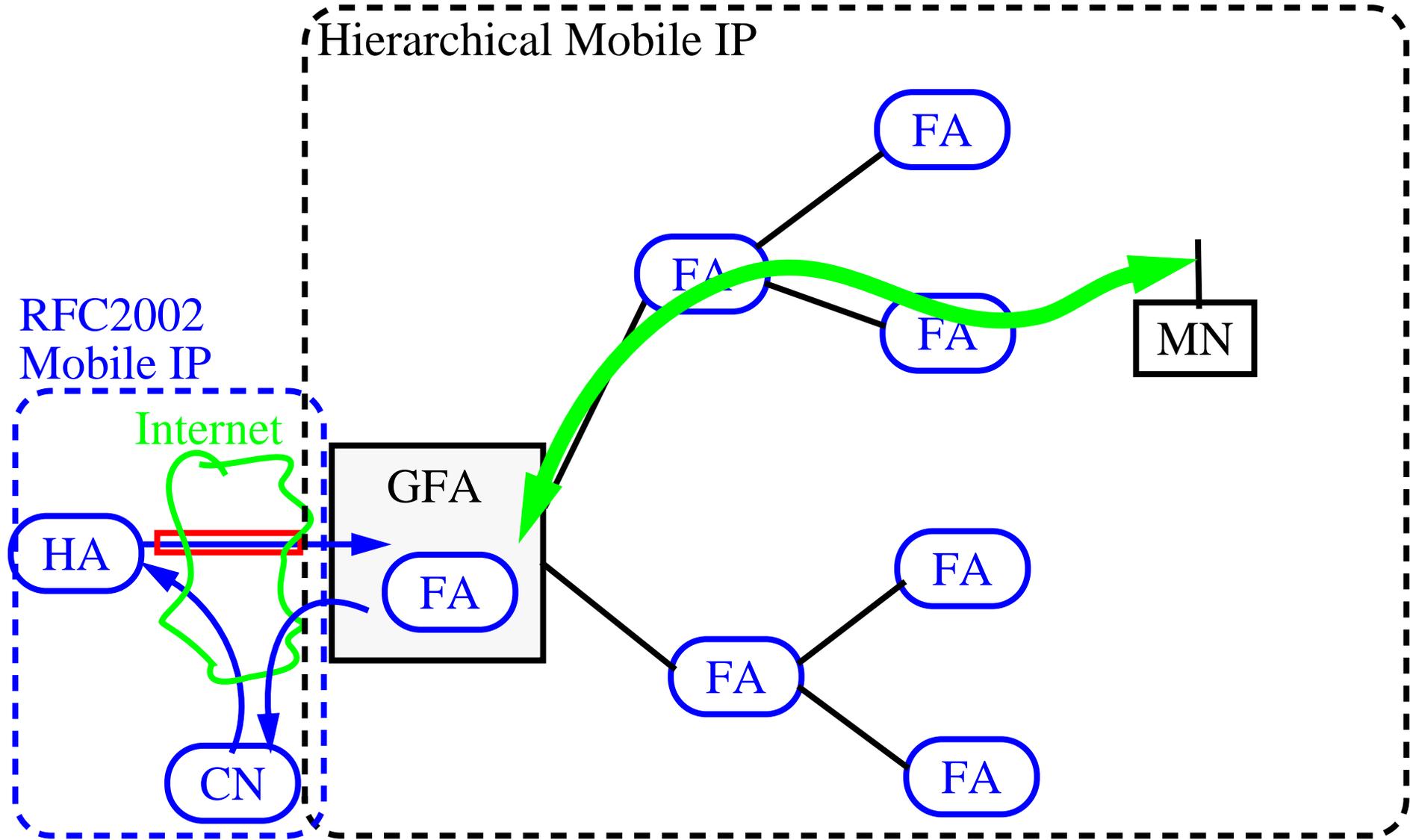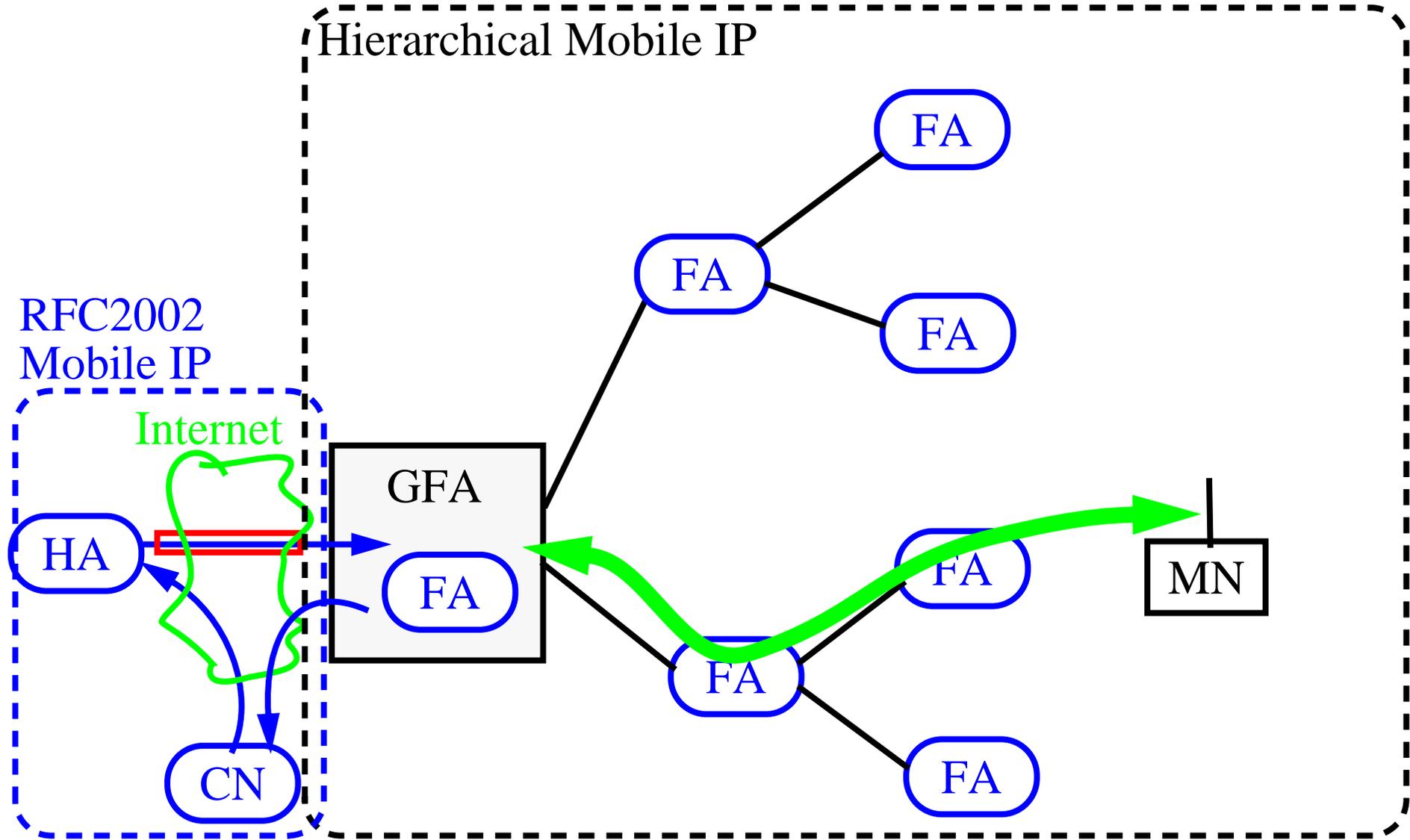
Why not simply use Dynamic DNS (DDNS)?

Problems of Dynamic DNS Mobility

• Only support inter-session mobility.

• TCP has to be disconnected when changing net.

• No inter-networking handover.

• Performance limitation problems.

• Security, Intranet firewall, etc.

	Mobile IP	Dynamic DNS
TCP survive the movement	Yes	No
Intra-session mobility	Yes	No
Handover Support	(Working on)	No
Performance Limitation	No	Yes

Thus DDNS does not really provide mobility, just connecting at different places.

Maguire
maguire@it.kth.se

Why not simply use Dynamic DNS (DDNS)?
2006.03.13

Mobile IP 661 of 663
Internetworking/Internetteknik

Summary

This lecture we have discussed:

- Mobile IP

Maguire
maguire@it.kth.se

Summary
2006.03.13

Mobile IP 662 of 663
Internetworking/Internetteknik

References

[105]. B. Aboba and M. Beadles, "The Network Access Identifier", IETF RFC 2486, January 1999 `http://www.ietf.org/rfc/rfc2486.txt`

[106]C. Perkins, "IP Encapsulation within IP", IETF RFC 2003, October 1996

   `http://www.ietf.org/rfc/rfc2003.txt`

[107]C. Perkins, "'Minimal Encapsulation within IP", IETF RFC 2004, October 1996 `http://www.ietf.org/rfc/rfc2004.txt`

[108]Juan Caballero Bayerri and Daniel Malmkvist, Experimental Study of a Network Access Server for a public WLAN access network, M.S. Thesis, KTH/IMIT, Jan. 2002

Maguire
maguire@it.kth.se

References
2006.03.13

Mobile IP 663 of 663
Internetworking/Internetteknik

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 12: IPSec, VPNs, Firewalls, and NAT

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapters 26 and 28

KTH Information and Communication Technology

Outline

- IPSec, VPN, …
- Firewalls & NAT
- Private networks

Maguire
maguire@it.kth.se

Outline
2006.03.13

IPSec, VPNs, Firewalls, and NAT 665 of 695
Internetworking/Internetteknik

Private networks

Private Networks are designed to be used by a limited set of users (generally those inside an organization)

Intranet	a private network - access limited to those in an organization
Extranet	intranet + limited access to some resource by additional users from outside the organization

Addresses for Private IP networks

- these should never be routed to outside the private network
- they should never be advertised (outside the private network)
- allocated (**reserved**) addresses:

Range	Total addresses
10.0.0.0 to 10.255.255.255	2^{24}
172.16.0.0 to 172.31.255.255	2^{20}
192.168.0.0 to 192.168.255.255	2^{16}

Maguire
maguire@it.kth.se
Private networks
2006.03.13
IPSec, VPNs, Firewalls, and NAT 666 of 695
Internetworking/Internetteknik

Virtual Private networks (VPNs)
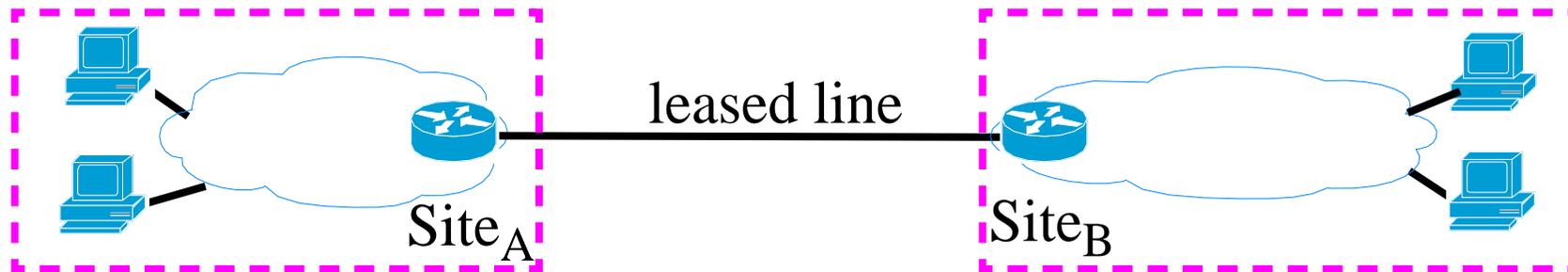
Figure 112: Private network
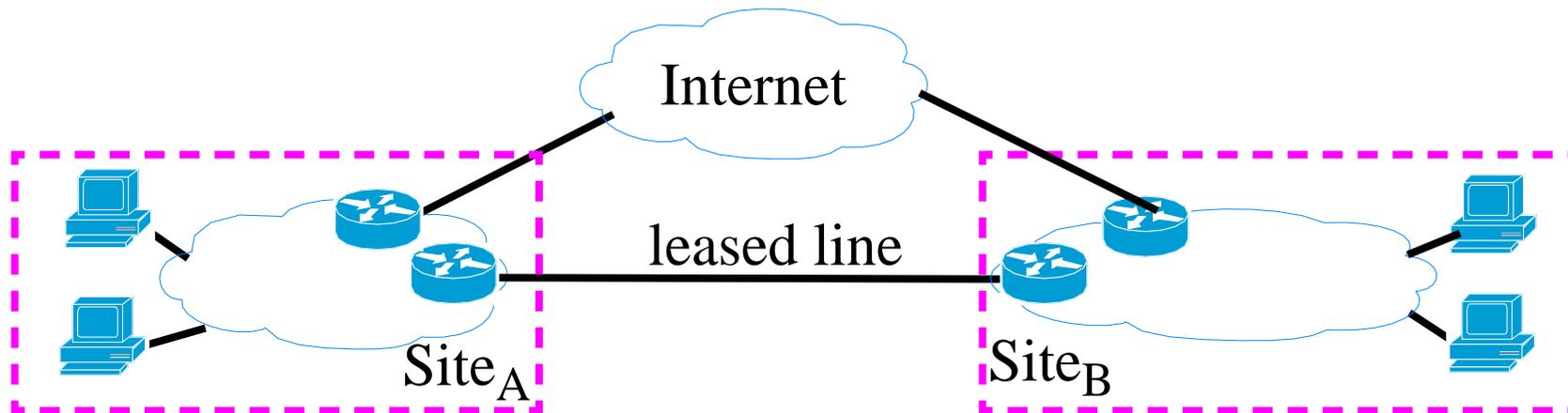
Figure 113: Hybrid network
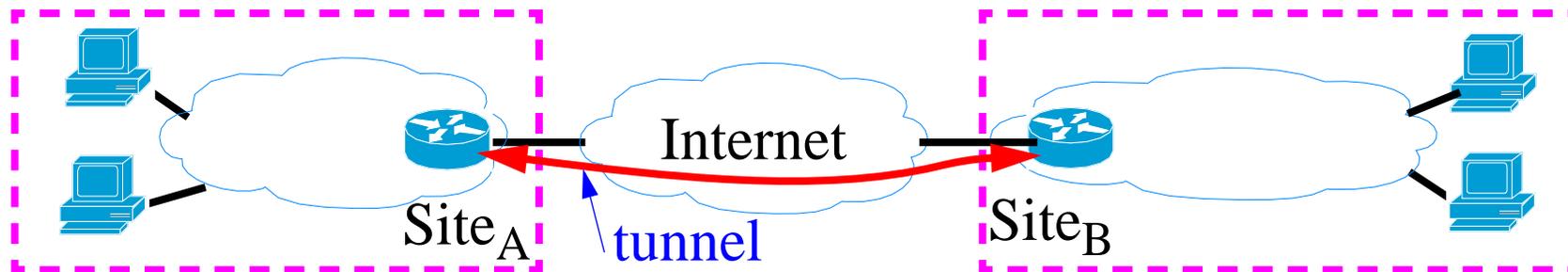
Figure 114: Virtual Private network

Maguire
maguire@it.kth.se

Virtual Private networks (VPNs)
2006.03.13

IPSec, VPNs, Firewalls, and NAT 667 of 695
Internetworking/Internetteknik

Security Protocols, APIs, etc.

- Generic Security Services App. Programming Interface (GSS-API)

- Network layer security

  - Internet Protocol Security Protocol (IPSEC)

- Secured Socket Layer (SSL)/Transport Layer Security

  - transport layer security
  - Secured HyperText Transport Protocol (S-HTTP)

- Application layer security

  - Pretty Good Privacy (PGP) [130]
  - Privacy-Enhanced Electronic Mail (PEM), S/MIME (signed MIME), PGP/MIME, and OpenPGP, … [131]
  - MasterCard and Visa's Secured Electronic Transaction (SET)

- Authentication

  - Remote Authentication Dial-In User Services (RADIUS)
    http://www.gnu.org/software/radius/radius.html, FreeRADIUS *http://www.freeradius.org/*
  - DIAMETER *http://www.diameter.org/*

- …

Maguire
maguire@it.kth.se
Security Protocols, APIs, etc.
2006.03.13
IPSec, VPNs, Firewalls, and NAT 668 of 695
Internetworking/Internetteknik

GSS-API

Generic Security Services Application Programming Interface (GSS-API)

- provides an abstract interface which provides security services for use in distributed applications
- but isolates callers from specific security mechanisms and implementations.

GSS-API peers establish a common security mechanism for security context establishment either through administrative action, or through negotiation.

GSS-API is specified in:

- J. Linn, "Generic Security Service API v2", RFC 2078 [116]
- J. Wray, "Generic Security Service API v2: C-bindings", RFC 2744 [117].

Maguire
maguire@it.kth.se

GSS-API
2006.03.13

IPSec, VPNs, Firewalls, and NAT 669 of 695
Internetworking/Internetteknik

IPSec

IPSec in three parts:

- encapsulating security payload (ESP) defines encryption or IP payloads,
- authentication header (AH) defines authentication method, and
- the IP security association key management protocol (ISAKMP) manages the exchange of secret keys between senders and recipients of ESP or AH packets.

Maguire
maguire@it.kth.se

IPSec
2006.03.13

IPSec, VPNs, Firewalls, and NAT 670 of 695
Internetworking/Internetteknik

ESP packet

Consists of:

- a control header - contains a Security Parameters Index (SPI) and a sequence number field (the SPI + destination IP address unqiuely identifies the Security Association (SA)).

- a data payload - encrypted version of the user's original packet. It may also contain control information needed by the cryptographic algorithms (for example DES needs an initialization vector (IV)).

- an optional authentication trailer - contains an Integrity Check Value (ICV) - which is used to validate the authenticity of the packet.

ESP could use any one of several algorithms: DES, Triple DES, …

See: RFC 2406: IP Encapsulating Security Payload (ESP)[110]

Maguire
maguire@it.kth.se

ESP packet
2006.03.13

IPSec, VPNs, Firewalls, and NAT 671 of 695
Internetworking/Internetteknik

AH header

For authentication purposes only contains:

- an SPI,
- a sequence number, and
- an authentication value.

AH uses either:

- Message Digest 5 (MD5) algorithm,
- Secure Hash Algorithm 1 (SHA-1),
- truncated HMAC (hashed message authentication code), or
- …

For further information see:

- IP Authentication Header - RFC 2402 [111]

Maguire
maguire@it.kth.se

AH header
2006.03.13

IPSec, VPNs, Firewalls, and NAT 672 of 695
Internetworking/Internetteknik

ISAKMP

ISAKMP is based on the Diffie-Hellman key exchange protocol; it assumes the identities of the two parties are known.

Using ISAKMP you can:

- control the level of trust in the keys,
- force SPIs to be changed at an appropriate frequency,
- identify keyholders via digital certificates
  [requires using a certificate authority (CA)]

For further information see:

- Internet Security Association and Key Management Protocol (ISAKMP)
  - RFC 2408 [112]
- The Internet IP Security Domain of Interpretation for ISAKMP - RFC 2407 [113]
- The OAKLEY Key Determination Protocol - RFC 2412 [114]
- The Internet Key Exchange (IKE) - RFC 2409 [115]

Maguire
maguire@it.kth.se

ISAKMP
2006.03.13

IPSec, VPNs, Firewalls, and NAT 673 of 695
Internetworking/Internetteknik

Where can you run IPSec?

Mode	Where it runs	Payload
Transport	end-systems	payload data follows the normal IP header
Tunnelling	internetworking device: e.g., router, firewall, or VPN gateway	• end-user's entire packet-IP headers and all-placed within another packet with ESP or AH fields [thus it is encapsulated in another packet] • can hide the original source and destination address information
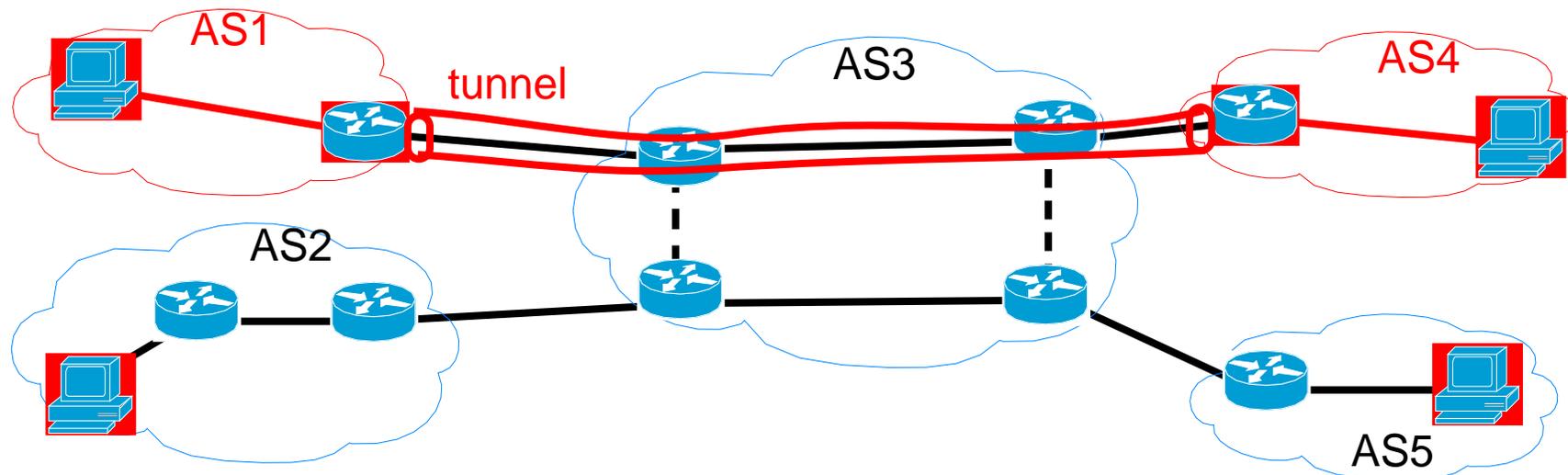
Figure 115: IPSec usage
red = secure, black = unsecure

Maguire
maguire@it.kth.se

Where can you run IPSec?
2006.03.13

IPSec, VPNs, Firewalls, and NAT 674 of 695
Internetworking/Internetteknik

Firewalls

exterior   interior (often an **Intranet**)
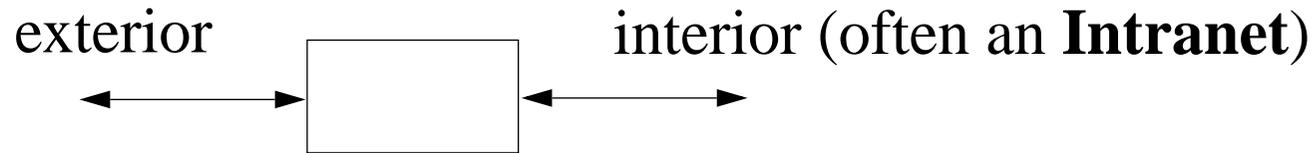
Figure 116: Firewall an internet gateway

The firewall can provide packet by packet filtering of packets coming into the **intranet** or leaving the intranet. The firewall can decide which packets should be forwarded based on source, destination addresses, and port (or even deeper examination) using an explicitly defined **policy**.

Maguire
maguire@it.kth.se

Firewalls
2006.03.13

IPSec, VPNs, Firewalls, and NAT 675 of 695
Internetworking/Internetteknik

Linux firewall

For example, for the software firewall used in Linux systems called "ipfwadm":

- all ports are typically closed for inbound traffic,

- all outbound traffic is "IP masqueraded", i.e., appears to come from the gateway machine; and

- For bi-directional services required by the users, "holes" may be punched through the firewall - these holes can reroute traffic to/from particular ports:
  - to specific users or
  - the most recent workstation to request a service.

Maguire
maguire@it.kth.se

Linux firewall
2006.03.13

IPSec, VPNs, Firewalls, and NAT 676 of 695
Internetworking/Internetteknik

Firewall Design

apply basics of security:

- **least privilege:**
  - don't make hosts do more than they have to (implies: specialize servers)
  - use minimum privileges for the task in hand
- **fail safe**
  - even if things break it should not leave anything open
- **defence in depth**
  - use several discrete barriers - don't depend on a single firewall for all security
- **weakest links**
  - know the limitations of your defences - understand your weakest link

Firewalls should have sufficient performance to keep the pipes full - i.e., a firewall should not limit the amount of traffic flowing across the connection to the external network, only **what** flows across it!

Maguire
maguire@it.kth.se

Firewall Design
2006.03.13

IPSec, VPNs, Firewalls, and NAT 677 of 695
Internetworking/Internetteknik

Proxy Access Through A Firewall

Bastion host

exterior   interior

Internet   Proxy Server   Intranet

manually enabled bypass
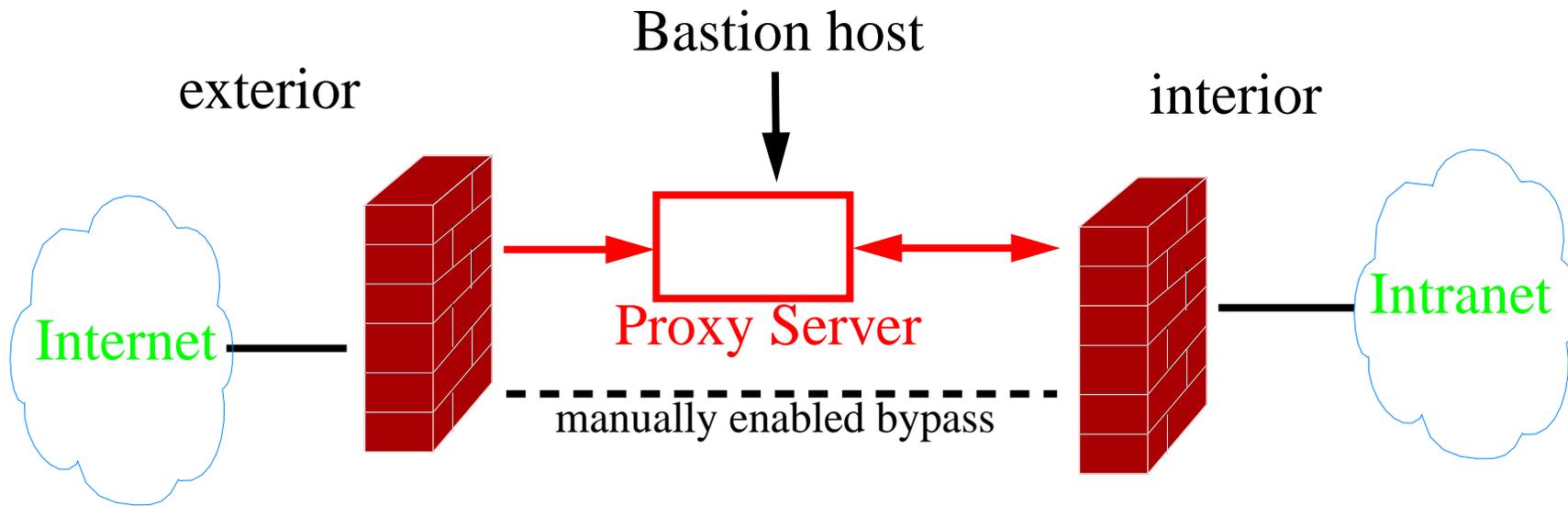
Figure 117: Firewall and internet gateway

Often you need application level proxies (i.e., they undertand details of the application protocol) -- an example is to proxy RealAudio's streaming audio.

SOCKs

Permeo Technologies, Inc.'s SOCKS *http://www.socks.nec.com/*

In order to bridge a firewall we can use a proxy:

- the proxy will appear to be **all external hosts** to those within the firewall
  - for example, If a user attached to the intranet requests a webpage, the request is sent to the proxy host where the same request is duplicated and sent to the real destination. When data is returned the proxy readdresses (with the user's intranet address) the returned data and sends it to the user.

- widely used to provide proxies for commonly used external services (such as Telnet, FTP, and HTTP).

See: [124] and [125]
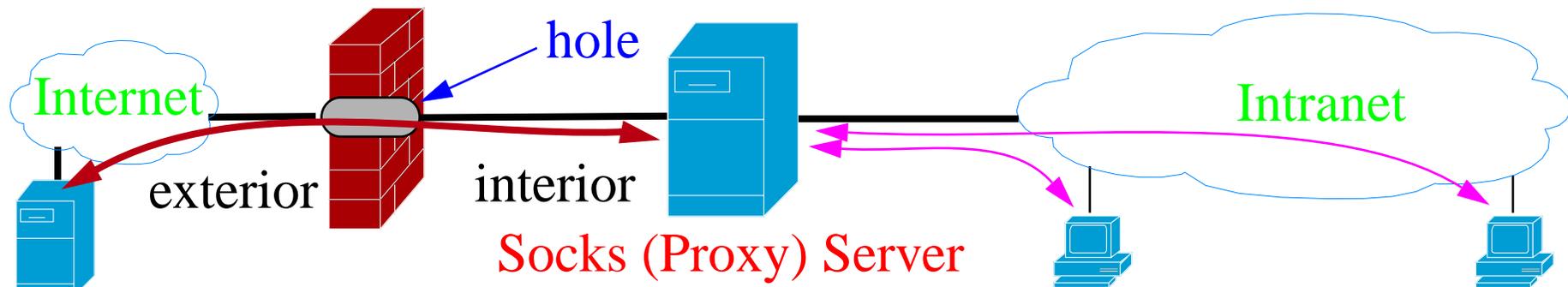
Figure 118: Firewall and internet gateway

Maguire
maguire@it.kth.se

SOCKs
2006.03.13

IPSec, VPNs, Firewalls, and NAT 679 of 695
Internetworking/Internetteknik

Newping

http://ftp.cerias.purdue.edu/pub/tools/dos/socks.cstc/util/newping.c

- a "ping" for SOCKS

- it depends on the target host **not** blocking the service on the appropriate port (in this case "**time**"). This version is primarily for checking "Is it alive?" rather than gathering statistics on the average response time of several echo requests.

- Uses the "**time**" TCP port to verify that a host is up, rather than using ICMP $\Rightarrow$ usable through a firewall that blocks ICMP.

Maguire
maguire@it.kth.se

Newping
2006.03.13

IPSec, VPNs, Firewalls, and NAT 680 of 695
Internetworking/Internetteknik

MBONE through firewalls

Their firewall features:

- Source host checking (allowing only certain hosts to transmit through the firewall, or denying specific hosts)
- Destination port checking
- Packet contents (unwrapping encapsulated IP)
- Regulating bandwidth allocated to a specific multicast group's traffic

Their Mbone gateway is based on a modified multicast routing daemon.

MBONE   hole   Intranet

join   join   join

SOCKS+Mrouted-gw
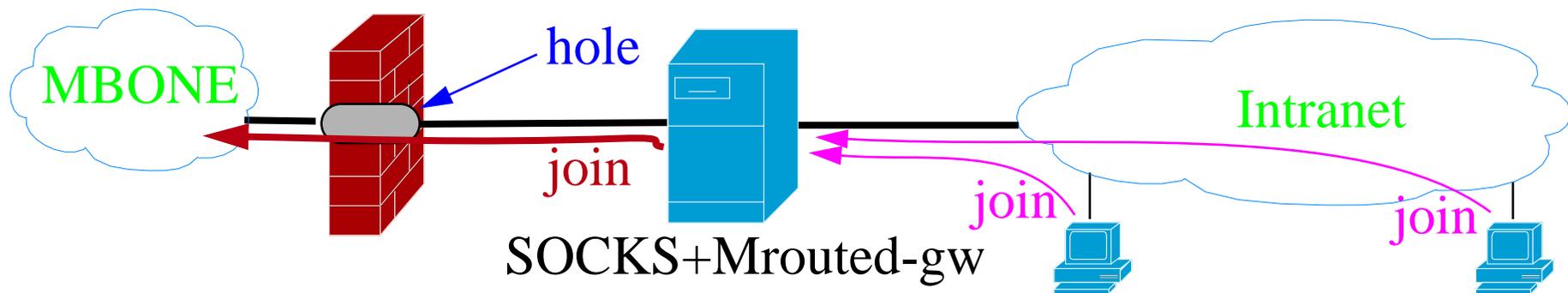
Figure 119: Firewall and internet gateway

Maguire
maguire@it.kth.se

MBONE through firewalls
2006.03.13

IPSec, VPNs, Firewalls, and NAT 681 of 695
Internetworking/Internetteknik

Secure Mailer (aka Postfix)

Wietse Venema's attempt to provide an alternative to the widely-used Sendmail program

70% of all mail sent via the Internet is sent via Sendmail

"Security. Postfix uses multiple layers of defense to protect the local system against intruders. Almost every Postfix daemon can run in a chroot jail with fixed low privileges. There is no direct path from the network to the security-sensitive local delivery programs - an intruder has to break through several other programs first. Postfix does not even trust the contents of its own queue files, or the contents of its own IPC messages. Postfix avoids placing sender-provided information into shell environment variables. Last but not least, no Postfix program is set-uid." [126]

Maguire
maguire@it.kth.se

Secure Mailer (aka Postfix)
2006.03.13

IPSec, VPNs, Firewalls, and NAT 682 of 695
Internetworking/Internetteknik

U.S. DOE CIAC's Network Security Tools [127]

- System Administrator Tool for Analyzing Networks (**SATAN**), network security analyzer designed by Dan Farmer and Wietse Venema; scans systems connected to the network noting the existence of well known, often exploited vulnerabilities. (see also Security Auditor's Research Assistant (SARA))

- **ipacl** - forces all TCP and UDP packets to pass through an access control list facility

- **logdaemon** - modified versions of rshd, rlogind, ftpd, rexecd,login, and telnetd that log significantly more information -- enabling better auditing of problems via the logfiles

- improved versions of: portmap, rpcbind,

- **screend** - a daemon and kernel modifications to allow all packets to be filtered based on source address, destination address, or any other byte or set of bytes in the packet

- **securelib** - new versions of the accept, recvfrom, and recvmsg networking system calls

Maguire
maguire@it.kth.se

U.S. DOE CIAC's Network Security Tools [127] IPSec, VPNs, Firewalls, and NAT 683 of
2006.03.13
Internetworking/Internetteknik

- **TCP Wrappers** - allows monitoring and control over who connects to a host's TFTP, EXEC, FTP, RSH, TELNET, RLOGIN, FINGER, and SYSTAT ports + a library so that other programs can be controlled and monitored in the same fashion
- **xinetd** - a replacement for inetd which supports access control based on the address of the remote host and the time of access + provides extensive logging capabilities

Maguire
maguire@it.kth.se

U.S. DOE CIAC's Network Security Tools [127] IPSec, VPNs, Firewalls, and NAT 684 of
2006.03.13
Internetworking/Internetteknik

The Network Mapper (NMAP)

Network Mapper (NMAP) *http://www.insecure.org/nmap/*

- (cleverly) uses raw IP packets
- determine what hosts are available on the network,
- what services (application name and version) are offered,
- what operating systems (and OS versions) they are running,
- what type of packet filters/firewalls are in use,
- …

http://www.insecure.org/nmap/nmap_documentation.html also has a link to "*Remote OS detection via TCP/IP Stack FingerPrinting*" by Fyodor <fyodor@dhp.com> (www.insecure.org), October 18, 1998 - a means of identifying which OS the host is running by noting its TCP/IP behavior.

Maguire
maguire@it.kth.se

The Network Mapper (NMAP)
2006.03.13

IPSec, VPNs, Firewalls, and NAT 685 of 695
Internetworking/Internetteknik

Network Address Translation
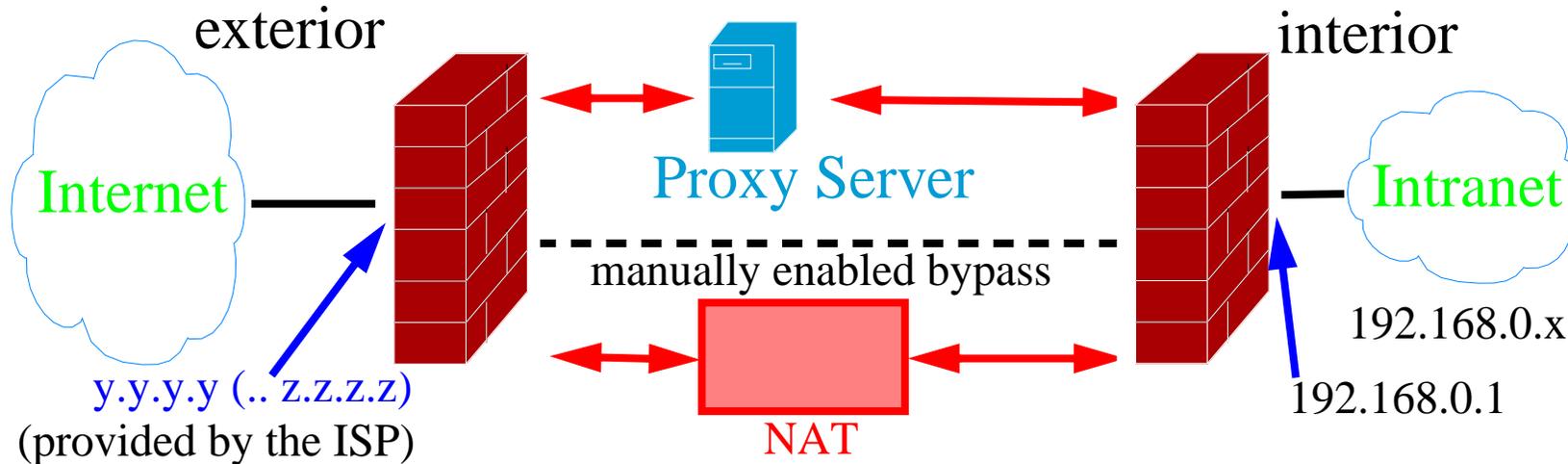
Figure 120: Example of a Firewall with NAT

NAT maps IP addresses on the inside to one or more addresses on the outside and vice versa. See RFC 3022 [137] and RFC2766 [138]

Advantages:	**Disadvantage**
✔ save IPv4 addresses	✘ Unfortunately this breaks many services because they use an IP address inside the their data.
✔ hides internal node structure from outside nodes	
✔ the intranet does not have to be renumbered when you connect to another ISP	

Maguire
maguire@it.kth.se

Network Address Translation
2006.03.13

IPSec, VPNs, Firewalls, and NAT 686 of 695
Internetworking/Internetteknik

Demilitarized zone (DMZ)
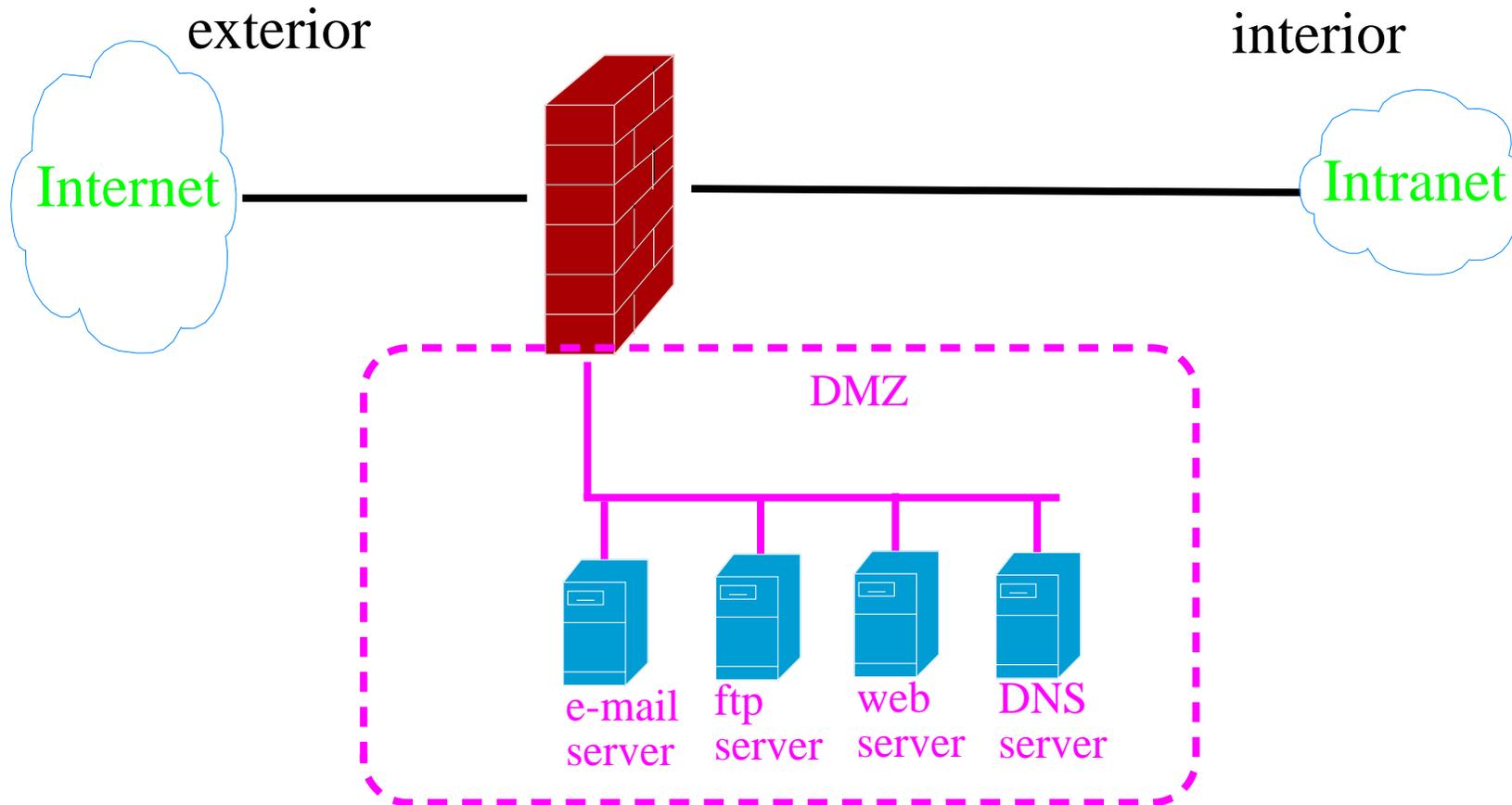
Figure 121: Example of a Firewall with a DMZ

Note that the various services may also be in different DMZ (see for example fogure 4 page 90 of [128]

Maguire
maguire@it.kth.se

Demilitarized zone (DMZ)
2006.03.13

IPSec, VPNs, Firewalls, and NAT 687 of 695
Internetworking/Internetteknik

Network Security Exercises

You will find a nice set of exercises by Ramesh Govindan at USC's ISI for Kerberos, S/Key, and firewalls at: `http://www.isi.edu/~govindan/cs558/netsec/index.html`

Note that you should **not** use their machines for these exercises, but I think you will find this useful reading.

Maguire
maguire@it.kth.se

Network Security Exercises
2006.03.13

IPSec, VPNs, Firewalls, and NAT 688 of 695
Internetworking/Internetteknik

Security Organizations and Companies

Computer Emergency Response Team (CERT [R]) Coordination Center [118]

- 1988 - Computer Emergency **Response** Team
- 2003 - Computer Emergency **Readiness** Team [122]

Addionally, there are numerous other CERTs:

- CanCERT™, GOVCERT.NL, Sveriges IT-incidentcentrum (SITIC) *http://www.sitic.se/*, Centre d'Expertise Gouvernemental de Réponse et de Traitement des Attaques informatiques (CERTA), CNCERT/CC [122], …
- The European Computer Security Incident Response Team Network *http://www.ecsirt.net/*

Forum of Incident Response and Security Teams (FIRST), now: 170 members[119]

NIST Computer Security Resource Center [120], Swedish Defense Material Administration, Electronics Systems Directorate [121], …

Maguire
maguire@it.kth.se

Security Organizations and Companies IPSec, VPNs, Firewalls, and NAT 689 of 695
2006.03.13
Internetworking/Internetteknik

Summary

This lecture we have discussed:

- Private networks
- IPSec
- Firewalls

Maguire
maguire@it.kth.se

Summary
2006.03.13

IPSec, VPNs, Firewalls, and NAT 690 of 695
Internetworking/Internetteknik

Further information

[109] IETF Security Area `http://sec.ietf.org/`

[110] S. Kent and R. Atkinson, "IP Encapsulating Security Payload (ESP)", IETF RFC 2406, November 1998 `http://www.ietf.org/rfc/rfc2406.txt`

[111] S. Kent and R. Atkinson, "IP Authentication Header", IETF RFC 2402, November 1998 `http://www.ietf.org/rfc/rfc2402.txt`

[112] D. Maughan, M. Schertler, M. Schneider, and J. Turner, "Internet Security Association and Key Management Protocol (ISAKMP)", IETF RFC 2408, November 1998 `http://www.ietf.org/rfc/rfc2408.txt`

[113] D. Piper, "The Internet IP Security Domain of Interpretation for ISAKMP", IETF RFC 2407, November 1998 `http://www.ietf.org/rfc/rfc2407.txt`

[114] H. Orman, "The OAKLEY Key Determination Protocol", IETF RFC 2412, November 1998 `http://www.ietf.org/rfc/rfc2412.txt`

[115] D. Harkins and D. Carrel, "The Internet Key Exchange (IKE)", IETF

Maguire
maguire@it.kth.se

Further information
2006.03.13

IPSec, VPNs, Firewalls, and NAT 691 of 695
Internetworking/Internetteknik

RFC 2409,November 1998 _http://www.ietf.org/rfc/rfc2409.txt_

[116]J. Linn, "Generic Security Service Application Program Interface, Version 2", IETF RFC 2078, January 1997, _http://www.ietf.org/rfc/rfc2078.txt_

[117]J. Wray, "Generic Security Service API Version 2 : C-bindings", IETF RFC 2744, January 2000 _http://www.ietf.org/rfc/rfc2744.txt_

[118] Computer Emergency Response Team _http://www.cert.org/_

[119]Forum of Incident Response and Security Teams _http://www.first.org/_

[120]U. S. National Institute of Standards and Technology (NIST), Computer Security Division, Computer Security Resource Center _http://csrc.nist.gov/_

[121]Swedish Defense Material Administration _http://www.fmv.se/_

[122]David Crochemore, "Response/Readiness: What R the new CERTS?", National Computer network Emergency Response technical Team/Coordination Center of China (CNCERT/CC) 2005 Annual Conference, Guilin, P.R.China, 30 March 2005

Maguire
maguire@it.kth.se

Further information
2006.03.13

IPSec, VPNs, Firewalls, and NAT 692 of 695
Internetworking/Internetteknik

http://www.cert.org.cn/upload/2005AnnualConferenceCNCERT/1MainConference/10.DavidCrochemore-NGCERTOI.pdf

[123]Centre d'Expertise Gouvernemental de Réponse et de Traitement des Attaques informatiques (CERTA) *http://www.certa.ssi.gouv.fr/*

[124]M. Leech, M. Ganis, Y. Lee, R. Kuris, D. Koblas, and L. Jones, "SOCKS Protocol Version 5", IETF RFC 1928, March 1996

http://www.ietf.org/rfc/rfc1928.txt

[125]P. McMahon, "GSS-API Authentication Method for SOCKS Version 5", IETF RFC 1961, June 1996 *http://www.ietf.org/rfc/rfc1961.txt*

[126] Postfix *http://www.postfix.org*

[127]U.S. DOE's Computer Incident Advisory Capability

http://ciac.llnl.gov/ciac/ToolsUnixNetSec.html

[128]Robert Malmgren, *Praktisk nätsäkerhet*, Internet Academy Press, Stockholm, Sweden, 2003, ISBN 91-85035-02-5

Maguire
maguire@it.kth.se
Further information
2006.03.13
IPSec, VPNs, Firewalls, and NAT 693 of 695
Internetworking/Internetteknik

[129] Charlier Kaufman, Radia Perlman, and Mike Speciner, *Network Security: Private Communication in a PUBLIC World*, Prentice-Hall, 1995, ISBN 0-13-061466-1

[130] Simson Garfinkel, *PGP: Pretty Good Privacy*, O'Reilly & Associates, 1995 ISBN 1-56592-098-8

[131] Internet Mail Consortium, "S/MIME and OpenPGP", Oct 15, 2004

    http://www.imc.org/smime-pgpmime.html

Firewalls

[132] Bill Cheswick and Steve Bellovin, *Firewalls and Internet Security: Repelling the Wily Hacker*, Addison Wesley, 1994, ISBN: 0-201-63357-4

[133] D. Brent Chapman and Elizabeth Zwicky, *Building Internet Firewalls*, O'Reilly, 1995, ISBN: 1-56592-124-0

[134] Tony Mancill, *Linux Routers: A Primer for Network Administrators* Prentice-Hall, 2001, ISBN 0-13-086113-8.

Maguire
maguire@it.kth.se
Further information
2006.03.13
IPSec, VPNs, Firewalls, and NAT 694 of 695
Internetworking/Internetteknik

[135]Firewalls mailing list *http://www.isc.org/index.pl?/ops/lists/firewalls/*

[136]Computer Security Institute (CSI) at *http://www.gocsi.com/*

NAT

[137] P. Srisuresh and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", IETF RFC 3022, January 2001

http://www.ietf.org/rfc/rfc3022.txt

[138]G. Tsirtsis and P. Srisuresh, "Network Address Translation - Protocol Translation (NAT-PT)", IETF RFC 2766, February 2000

http://www.ietf.org/rfc/rfc2766.txt

Maguire
maguire@it.kth.se

Further information
2006.03.13

IPSec, VPNs, Firewalls, and NAT 695 of 695
Internetworking/Internetteknik

2G1305 Internetworking/Internetteknik Spring 2006, Period 4

Module 13: Future and Summary

Lecture notes of G. Q. Maguire Jr.

**KTH Information and
Communication Technology**

Outline

- QoS
- Interface trends
- IP SANs (Storage Area Networks): iSCSI, …
- A glimpse into the future.

Maguire
maguire@it.kth.se

Outline
2006.03.13

Future and Summary 697 of 740
Internetworking/Internetteknik

Quality of Service (QoS)

QoS refers to statistical performance guarantees that a network can make regarding packet loss, delay, throughput, and jitter.

Best effort delivery means no QoS guarantee.

QoS is thought to be more and more important this days.

Many proposals, implementations and studies.

Does Internet need QoS? How can IP network provide it?

Maguire
maguire@it.kth.se

Quality of Service (QoS)
2006.03.13

Future and Summary 698 of 740
Internetworking/Internetteknik

Service Differentiation

Integrated Services (InteServ):

- RSVP: connection request
- All nodes IntServ-capable
- Scalability
- Complicated network management

Differentiated Service (DiffServ): end of one-size-fits-all

- Classes of Service
- QoS based Routing
- Classes of Service at Gigabit rates
- New Pricing and Billing Policies
- New Resource Allocation Methods

    See: [139]

Maguire
maguire@it.kth.se

Service Differentiation
2006.03.13

Future and Summary 699 of 740
Internetworking/Internetteknik

Constraint-based Routing

QoS routing: selects network routes with sufficient resources for the requested QoS parameters

- to satisfy the QoS requirements for every admitted connection;
- to achieve network efficiency in resource ultilization.

Policy-based Routing: e.g. Virtual Private Networks (VPN)

How can we combine this with IP mobility?

Maguire
maguire@it.kth.se

Constraint-based Routing
2006.03.13

Future and Summary 700 of 740
Internetworking/Internetteknik

Performance

Routers:

1/2 to 1 Million packets per second (pps) for every gigabit per second of aggregate bandwidth

more than 250,000 routes

Maguire
maguire@it.kth.se

Performance
2006.03.13

Future and Summary 701 of 740
Internetworking/Internetteknik

PC interfaces

Standard I/O ports of PCs:

- Accelerated Graphics Port (AGP)

- PCI
  - Version 2.1 PCI bus - 64 bit, 66MHz, can burst to 528 Mbps
  - PCI-X 2.0: "High Performance, Backward Compatible PCI for the Future" [140]
  - PCI-X 533, offering up to 4.3 gigabytes per second of bandwidt

- Universal Serial Bus (USB)
  - USB: 12Mbps - with plug and play
  - USB 2.0 [141]

- Apple Computers' Firewire™ $\Rightarrow$ IEEE 1394
  - supporting more than 400 Mbps
  - P1394B (Gigabit 1394) defined in IEEE Std 1394b-2002

- 10/100/1000 Ethernet

Maguire
maguire@it.kth.se

PC interfaces
2006.03.13

Future and Summary 702 of 740
Internetworking/Internetteknik

Fibre Channel

From the X2T11 standards activity

Topologies: Point-to-Point, Fabric, and Arbitrated loop

Addresses: Loops, LANs, and worldwide addresses

Fibre Channel Profiles

Fibre Channel products

- Disk drives
- Network interfaces

Maguire
maguire@it.kth.se

Fibre Channel
2006.03.13

Future and Summary 703 of 740
Internetworking/Internetteknik

IP Storage Area Networks (SANs)

Using IP in conjunction with storage:

- Fibre Channel Over IP (FCIP)

JBOD == Just a Bunch of Disks
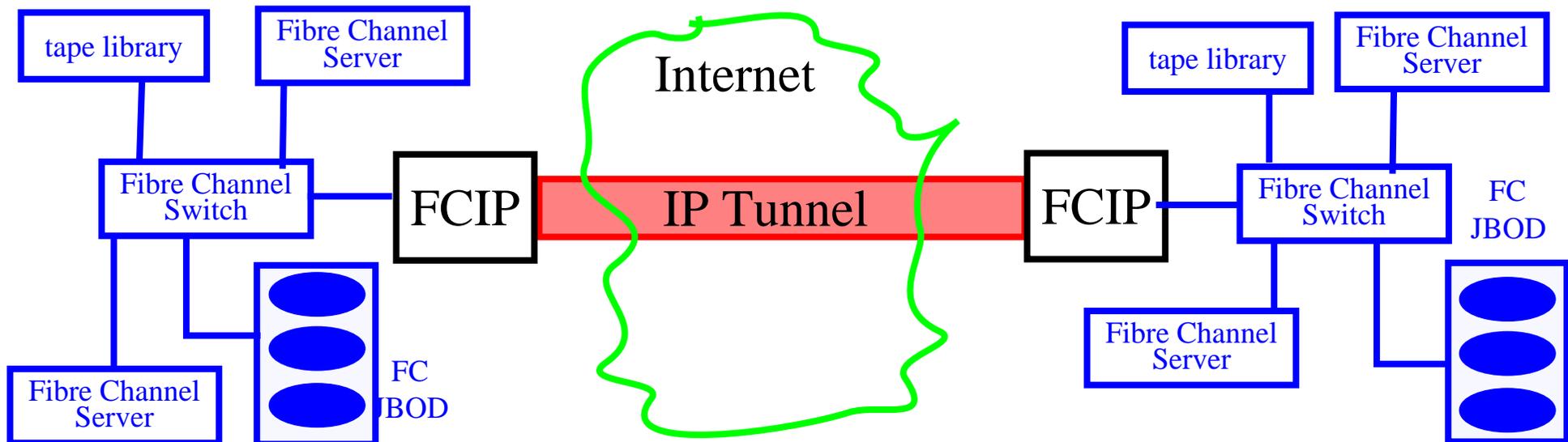
Note that this approach simply interconnects the two Fibre Channel switches. The connection between the two switches is TCP and it simply encapsulates a FCIP header and a Fibre Channel Frame.

Maguire
maguire@it.kth.se

IP Storage Area Networks (SANs)
2006.03.13

Future and Summary 704 of 740
Internetworking/Internetteknik

- ## Internet Fibre Channel Protocol (iFCP)

JBOD == Just a Bunch of Disks
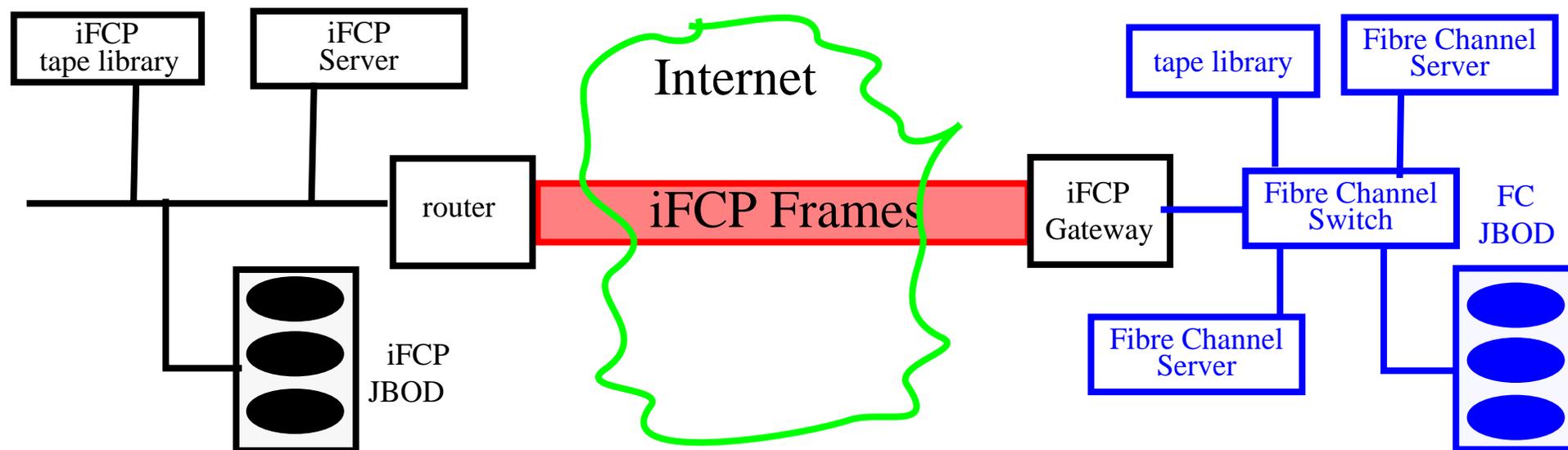
Note that this approach interconnects Fibre Channel devices. The connection between the two switches is TCP and it encapsulates a iFCP header and a Fibre Channel Frame; note that iFCP devices can simply be attached to the internet or an intranet. This means that there has to be a mapping between Fibre Channel addresses an IP addresses.

Maguire
maguire@it.kth.se

IP Storage Area Networks (SANs)
2006.03.13

Future and Summary 705 of 740
Internetworking/Internetteknik

- ## Internet SCSI (iSCSI)

JBOD == Just a Bunch of Disks
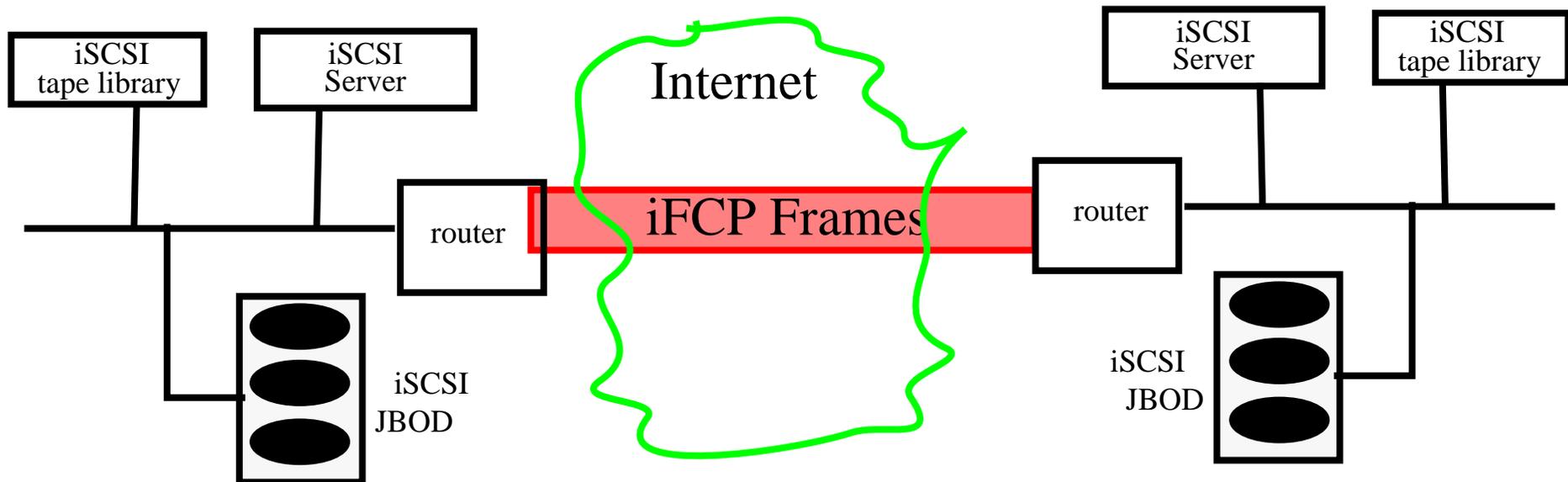
One a SCSI initiator has logged-in to a SCSI target, it can simply issue SCSI commands, just as if the device were on a local SCSI chain!

For more information see [142]

Maguire
maguire@it.kth.se

IP Storage Area Networks (SANs)
2006.03.13

Future and Summary 706 of 740
Internetworking/Internetteknik

Clustering

Myricom, Inc. *http://www.myri.com/*

American National Standards Institute (ANSI) Standard -- ANSI/VITA 26-1998

- Started by
  - Prof. Charles L. Seitz - Caltech, now President and CEO
  - Dr. Robert Felderman - Director of Software Development
  - Mr. Glenn Brown - Engineer and programmer

Clusters used to form high performance servers, using commodity networks and hosts. For performance numbers see: *http://www.myri.com/myrinet/overview/*

Maguire
maguire@it.kth.se

Clustering
2006.03.13

Future and Summary 707 of 740
Internetworking/Internetteknik

"Beowulf-class" machines

Using large numbers of commodity machines to make high performance computational systems by interconnecting them with a network.

- LANL's Loki *http://loki-www.lanl.gov*

- LANL's Avalon *http://cnls.lanl.gov/avalon/*

- JPL's Hyglac *http://hpc.jpl.nasa.gov/PS/HYGLAC/hyglac.html*

- INRIA's PopC (Pile of PCs)

- …

Maguire
maguire@it.kth.se

"Beowulf-class" machines
2006.03.13

Future and Summary 708 of 740
Internetworking/Internetteknik

Very high-speed Backbone Network Service (vBNS)

vBNS project (http://www.vbns.net/) created to provide a backbone for the US high-performance computing users and their SuperComputer Centers.

- mostly OC12C, but now adding OC48C links (2.4Gbps)
- connections to all NAPs
- provide for multimedia services (provides multicast)
- participate in developing advanced routing technologies
- supports IPv4 and IPv6

vBNS backbone network (*http://www.stanford.edu/group/itss-cns/i2/vbns.html*)

There is some interest in using vBNS for inter-gigapop interconnections.

Maguire
maguire@it.kth.se

Very high-speed Backbone Network Service (vBNS)
2006.03.13

Future and Summary 709 of 740
Internetworking/Internetteknik

Internet2

http://www.internet2.org/

- World class **research**
  Driven by computational physics, biology, chemistry, … and scientific visualization, virtual "experiments", and remote control of real experiments.

- Networking R&D - focused on exploiting the capabilities of broadband networks media integration, interactivity, real time collaboration, …

- Improve **production** Internet services and applications for all members of the academic community, both nationally and internationally.

Purpose: support national research objectives, distance education, lifelong learning, and related efforts.

http://www.hpcc.gov/white-house/internet/background.html

Maguire
maguire@it.kth.se

Internet2
2006.03.13

Future and Summary 710 of 740
Internetworking/Internetteknik

Gigapops

Who will be operating them?

Where will they be?

How many will there be?

What is the aggregate throughput that they will require?

What is the maximum per port throughput?

How many ports will they need to support?

Will they support "mixing"? (mixing is used to defeat traffic analysis)

Whose hardware and software will they use? What is the required functionality?

Maguire
maguire@it.kth.se

Gigapops
2006.03.13

Future and Summary 711 of 740
Internetworking/Internetteknik

Speed through Silicon

FPGAs used in many routers - for flexibility and to allow near hardware speed implementations of protocols.

ASICs: Vertex Networks, Inc. , MMC Networks, Inc, Galileo Technology, TI, …

Maguire
maguire@it.kth.se

Speed through Silicon
2006.03.13

Future and Summary 712 of 740
Internetworking/Internetteknik

Future networks

Terabit per second == 10^{12}

Readily achievable via combining multiple Gigabit per second streams using Wavelength Division Multiplexing (WDM).

Petabit per second== 10^{15}

Differentiated Services: Classes of Service, Multimedia

Constraint-based Routing (QoS Routing)

Ad Hoc Networking

Auto-configuration (Plug and Play Internet)

Active Networking

Smart Networking

Knowledge-based Networking

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 713 of 740
Internetworking/Internetteknik

Active Networks

- Network nodes can perform customized computations on the messages flowing through them.

- Can change, modify the contents of the messages.

- Potentially Mobility Enabling Routing using active network concept

Client — Active Node — Node — Active Node — Server

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 714 of 740
Internetworking/Internetteknik
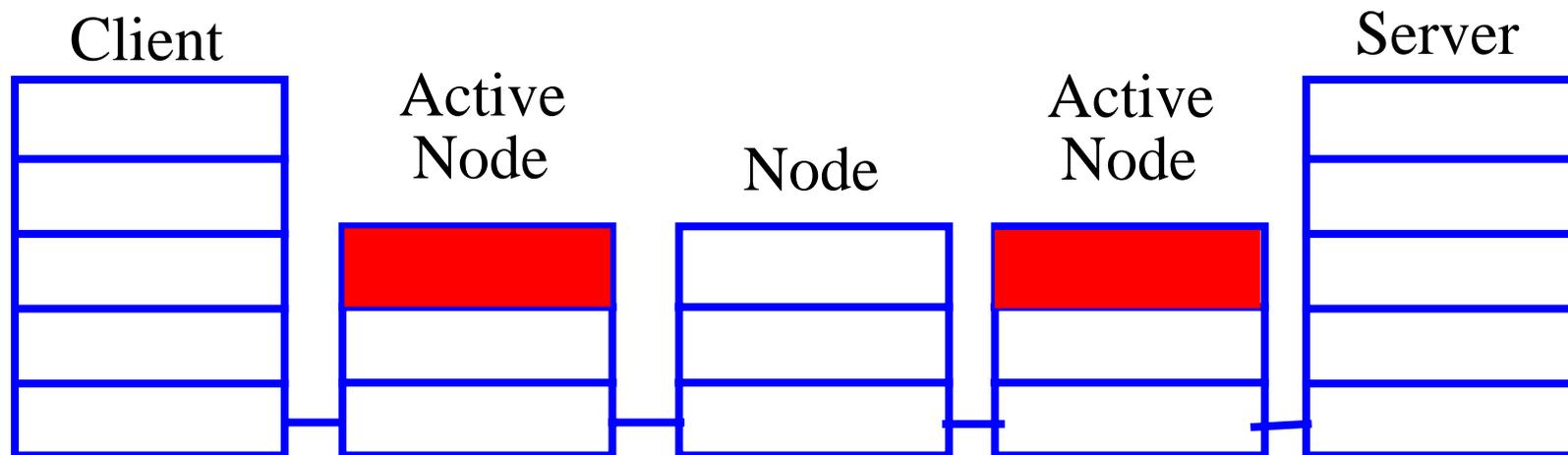
Smart Networks with Sensors

(Ren and Maguire, KTH)

- Context/situation-aware Systems
- Smart services: active + user-awareness networking
- Knowledge-based Networking

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 715 of 740
Internetworking/Internetteknik
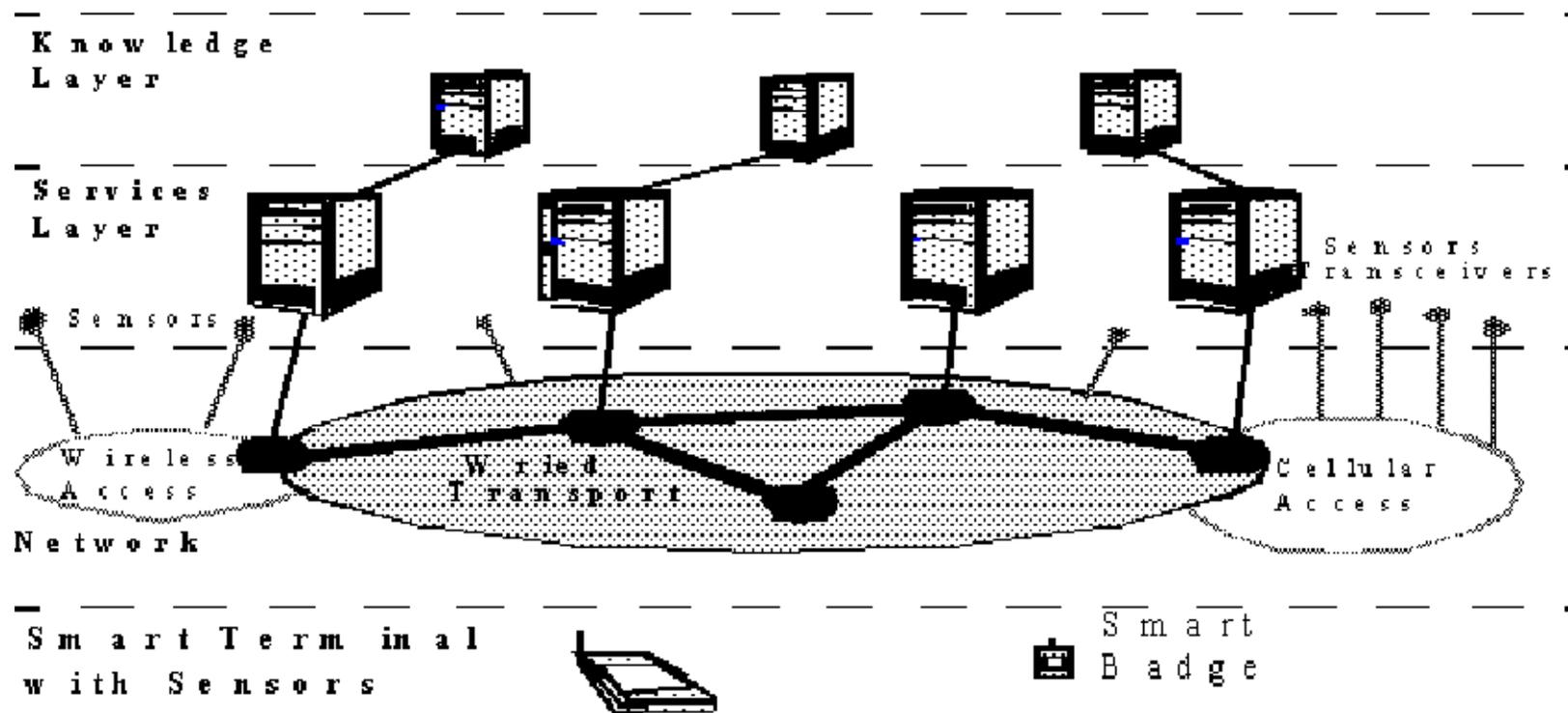
Bottlenecks

- Server and Network Bandwidth and latency

High speed networks

Low $\Rightarrow$ high speed wireless networks
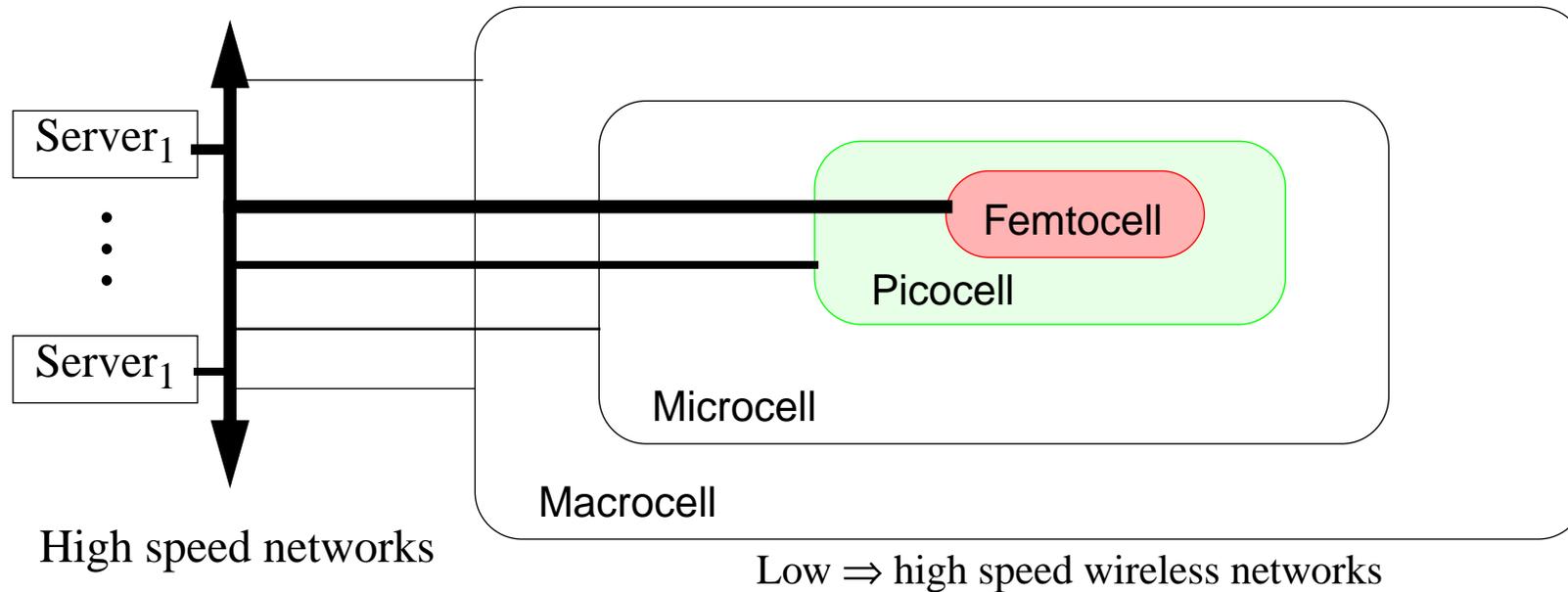
- User Bandwidth and latency
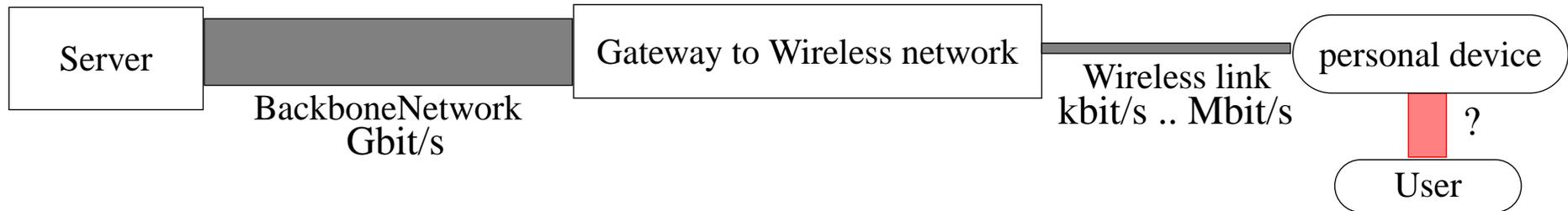
- Power and Energy $\Rightarrow$ need a computational theory of O(energy)

- Imagination!

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 716 of 740
Internetworking/Internetteknik

Near Future systems

Personal Portal
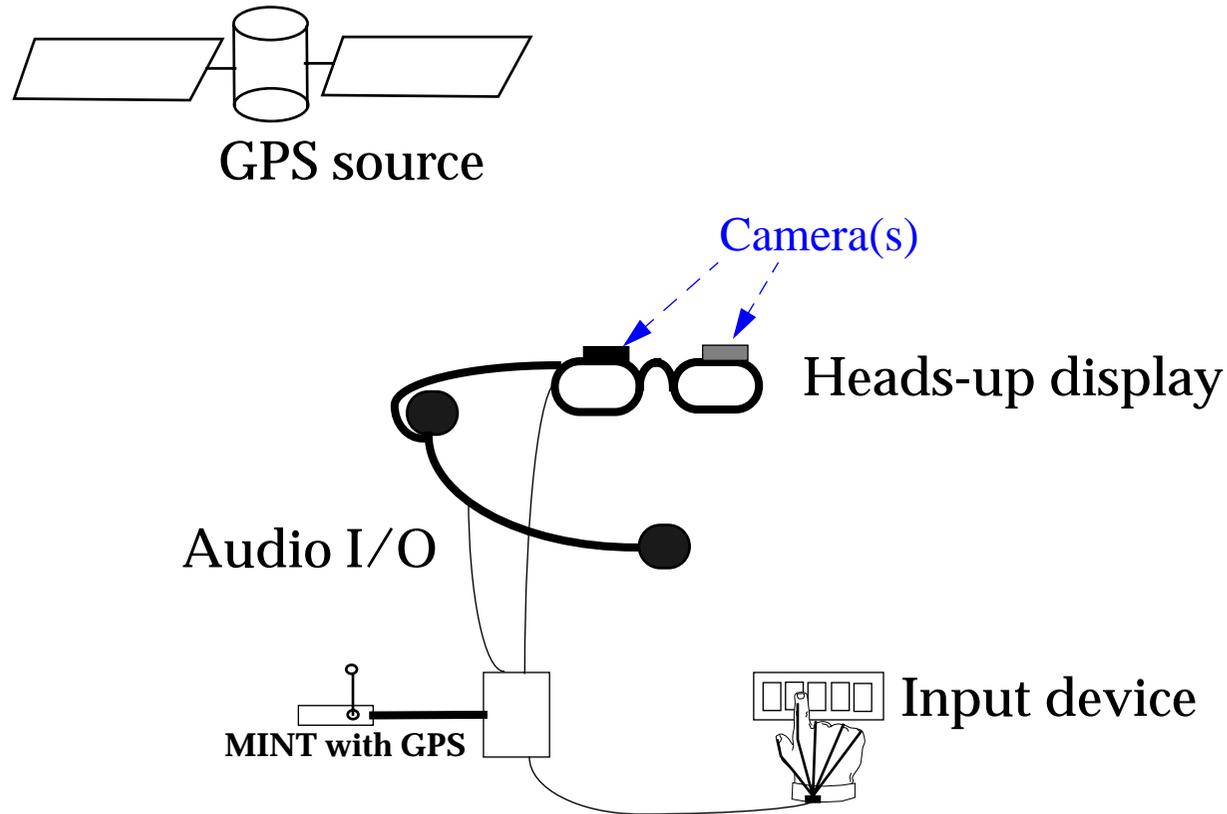
Figure 122: Vision-2, 2000 - high level of integration

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 717 of 740
Internetworking/Internetteknik

Evolution of new varieties of networks

Already we have: WANs (Wide Area), MANs (Metropolitan Area), LANs (Local Area Networks)

VANs   Vehicle Area Networks

Very local networks

DANs   Desk Area Networks

The computer/printer/telephone/… will all be part of a very local area network on your desk.

- wireless links ⇒ No longer will you have to plug your printer into your computer (PDA/…) into your computer
- active badges⇒ No longer will you have to sign in/out of areas, write down peoples names at meetings, … the system can provide this data based on the active badges

Olivetti and Xerox are exploring "Teleporting" your windows environment to the workstation nearest you, on command, if there are multiple choices probe each one (currently a "beep" is emitted to tell the user which).

BANs   Body Area Networks

Users will be carrying multiple devices which wish to communicate:

- thus there will be a need for a network between these devices which you carry around; and
- personal devices will wish to interact with fixed devices (such as Bankomat machines, vehicle control systems, diagnostic consoles (for a "mechanic" or repairman), …) and other peripherals.

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 718 of 740
Internetworking/Internetteknik
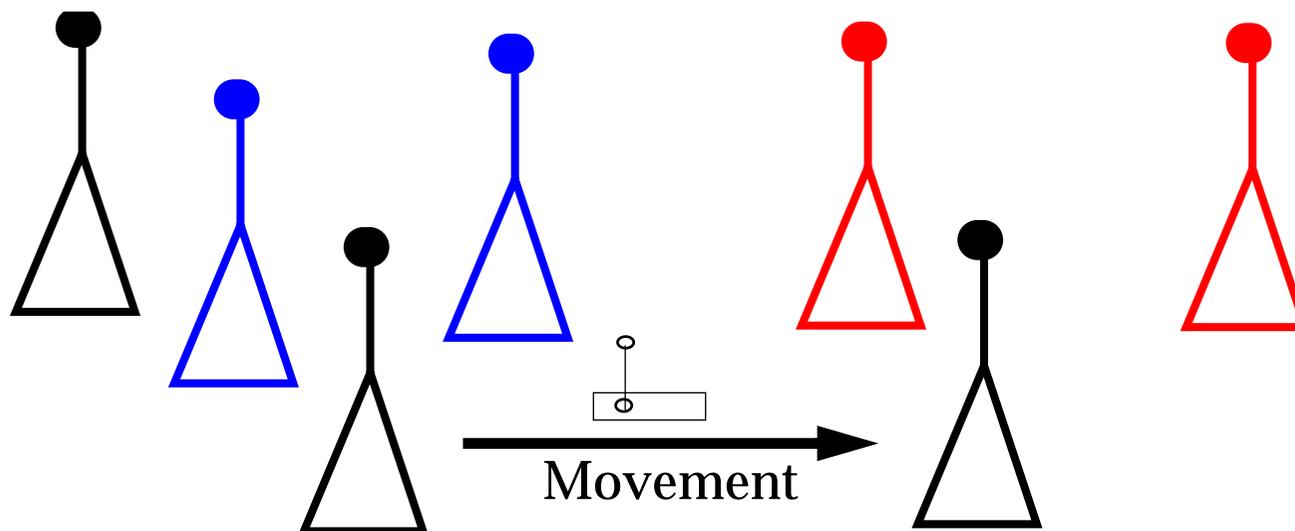
Situational awareness and Adaptability

Figure 123: Where am I? What am I? Who am I?
Where am I going? When will I be there? What should I become? Who should I become?

- Location dependent services

- Predicting location to reduce latency, reduce power, hide position, …

- Adapting the radio to the available mode(s), purposely changing mode, …

- Reconfigure the electronics to adapt, for upgrades, for fault tolerance, …; Reconfiguration vs. powering up and down fixed modules (what are the "right" modules, what is the "right" means of interconnect, what is the "right" packaging/connectors/…, needed speed of adaptation)

- "right" level of independence; spectrum from Highly Independent $\Rightarrow$ Very Dumb

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 719 of 740
Internetworking/Internetteknik

Location Dependent service(s)

How do I know where I am?

- Outdoors: GPS or from the network operators knowledge [resolution: 100m to sub-centimeter]
- Indoor: IR and RF beacons, triangulation, knowing what you can **see** or **hear**

What can I do with this knowledge?

KTH students built a JAVA Applet which gets data from GPS unit and dynamically displays a list of the information available - as a function of where you are:

- ♦ if near bus, subway, train stop - you get transit information - potentially with real-time schedule - since the system knows current location of vehicles
- ♦ list of restaurants, shops, etc. where you are and in the direction you are headed
  - ◆ the scope is based on your velocity vector - so if you move quickly it reduces detail, but increases the scope
- ♦ map information with updated position

How do I know who I'm with or what I'm near?

- Olivetti, Xerox, and MIT - using IR emitters as "ID" tags
  - ♦ Olivetti put them on people, equipment, …
  - ♦ Xerox put them on electronic notepads, rooms, …
  - ♦ MIT Media Lab is putting them on people + lots of inanimate objects (clock, fish tank, …)

Maguire
maguire@it.kth.se
Future networks
2006.03.13
Future and Summary 720 of 740
Internetworking/Internetteknik

Human centered

- Computer - human interaction is currently focused on the computer (computer-centric)
  - ◆ Currently computers know little about their environment
    - ◆ Where are we?
    - ◆ Who is using me?
    - ◆ Is the user still there?
- Evolving Environment awareness
  - ◆ Give computers senses via sensors
    - ◆ Environment
    - ◆ User identity and presence
- Badge as a smart card replacement
  - ◆ biometric signature of the person currently using the badge
  - ◆ the badge ensures that only you can use it
- You wear your own personal user interface
  - ◆ interface can be consistent across all appliances
    - ◆ not because each appliance supports the interface, but because the user's own interface provides consistency
- Make the human the focus of the computer's interaction ($\Rightarrow$ human-centric)

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 721 of 740
Internetworking/Internetteknik

Requirements

- Systems with which humans wish to interact:
  - ◆ traditional computers, desktop workspaces, domestic appliances, building and automotive systems, doors, elevators (lifts), environmental control, seats and mirrors, etc.

- Systems to provide sensor data:
  - ◆ location, orientation, light, heat, humidity, temperature, gas analysis, biomedical, …

- Systems to correlate the sensor information and provide it in a useful way to the computer systems:
  - ◆ Spatial and temporal sensor fusion,
  - ◆ 3D and 4D databases,
  - ◆ Machine Learning, and
  - ◆ Prediction (based on pattern extraction)

- Agents and actuators to provide intelligent control of the environment

- wireless/wired/mobile communications infrastructures to link it all together
  - ◆ must assure privacy and security

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 722 of 740
Internetworking/Internetteknik

Dumb Badge, Smart Badge, and Intelligent Badge

- Dumb Badge just emits its ID periodically

- Smart Badge - [an IP device] Location and Context Aware (i.e., a sensor platform)

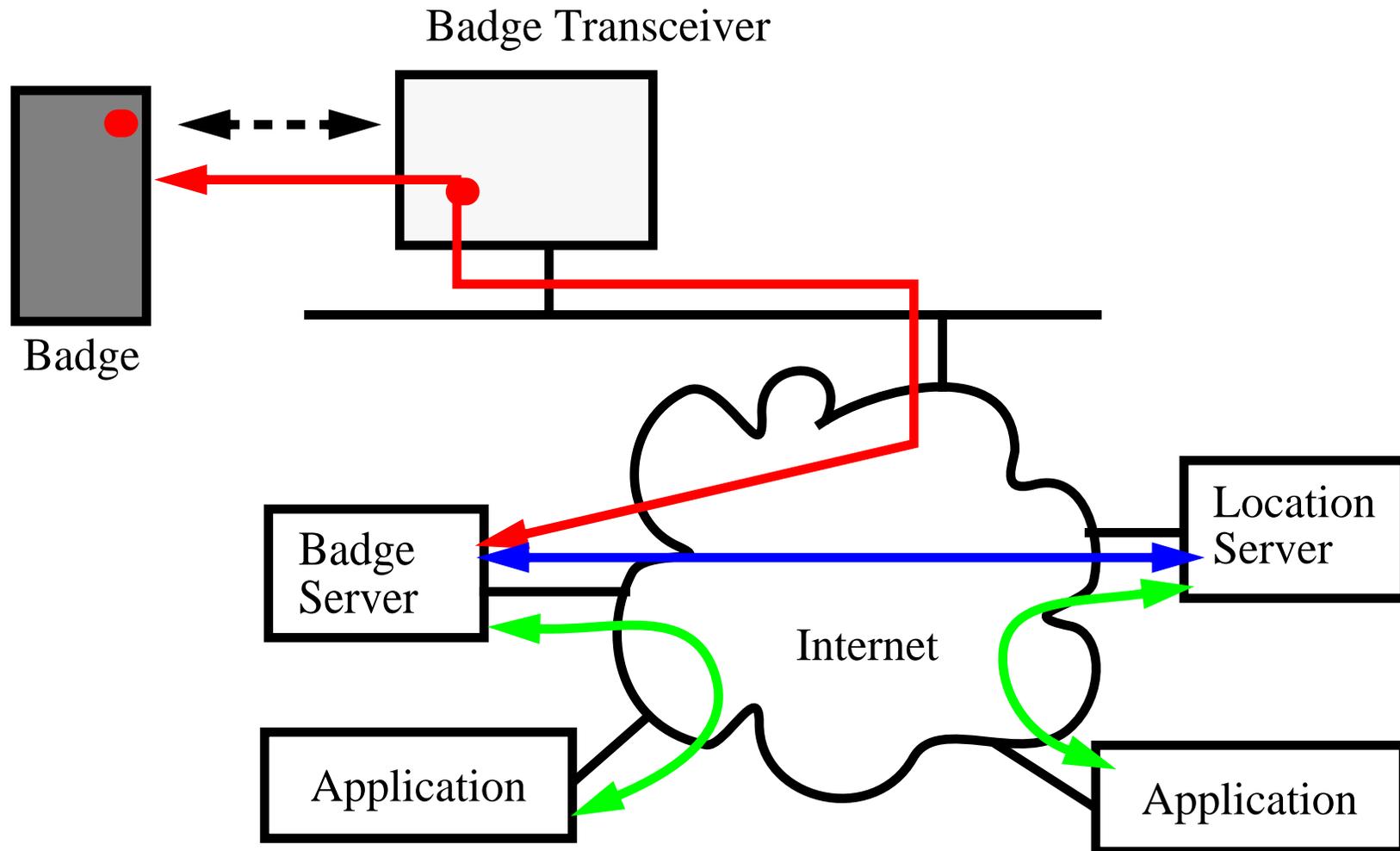- Intelligent Badge - add local processing for local interaction by the user

Acknowledgment:

All of the badge work is done in cooperation with:

- Dr. Mark T. Smith - Hewlett-Packard Research Laboratories, Palo Alto, California, USA

- Dr. H. W. Peter Beadle

  - ◆ Formerly: University of Wollongong, Wollongong, Australia
  - ◆ Currently: Director, Motorola Australian Research Centre, Botany, NSW, Australia
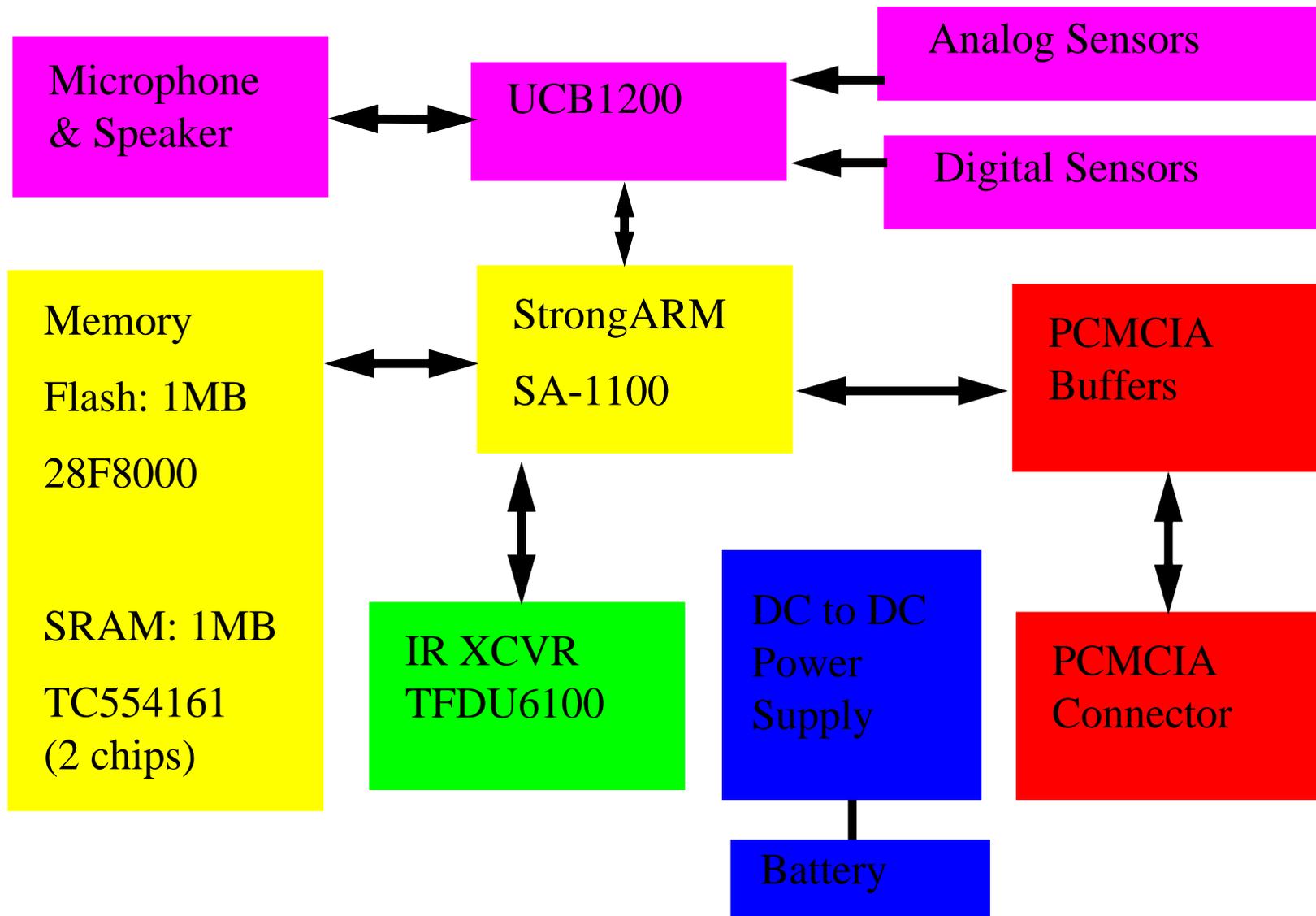
Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 723 of 740
Internetworking/Internetteknik

Badge Communications Model

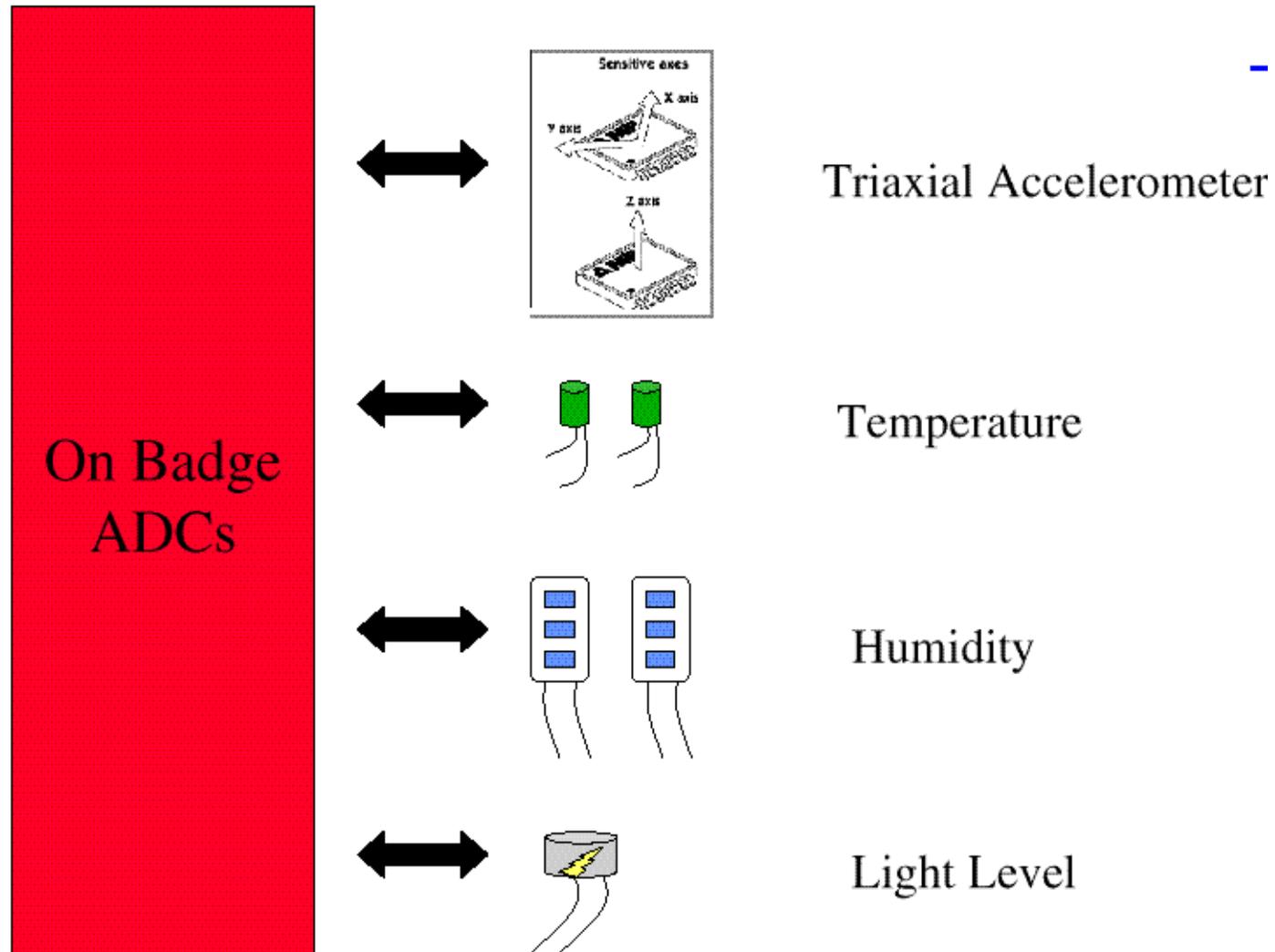Badges are IP devices (or should be), they communicate via network attached access points.

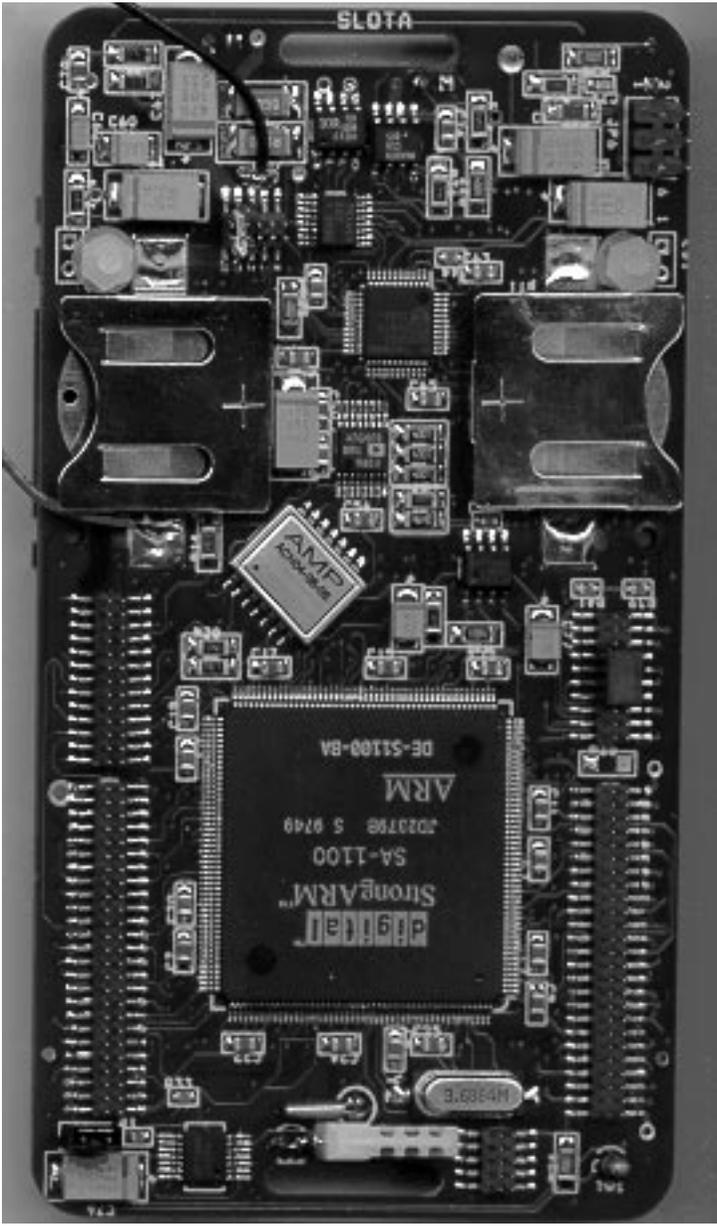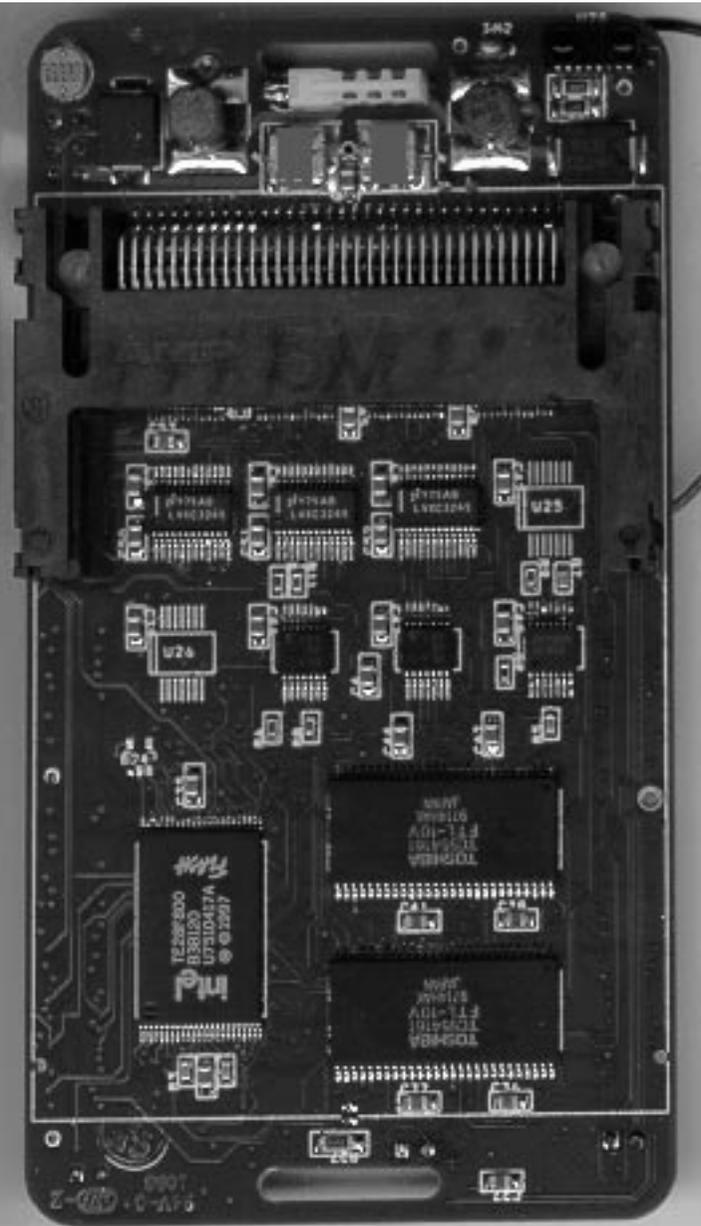$\Rightarrow$ Banks as intermediaries (**if** they have **any** future role)

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 724 of 740
Internetworking/Internetteknik

Smart Badge 3

Microphone & Speaker

UCB1200

Analog Sensors

Digital Sensors

Memory

Flash: 1MB

28F8000

SRAM: 1MB

TC554161 (2 chips)

StrongARM SA-1100

PCMCIA Buffers

IR XCVR TFDU6100

DC to DC Power Supply

PCMCIA Connector

Battery

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 725 of 740
Internetworking/Internetteknik

Smart Badge Sensors

On Badge ADCs

Triaxial Accelerometer

Temperature

Humidity

Light Level

Details of the 3rd version:
http://www.it.kth.se/edu/gru/Fingerinfo/telesys.finger/Mobile.VT98/badge3.html

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 726 of 740
Internetworking/Internetteknik

Badge 3

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 727 of 740
Internetworking/Internetteknik

A view of the packaged badge

As shown by HP at Comdex'98, November 16-20, 1998

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 728 of 740
Internetworking/Internetteknik

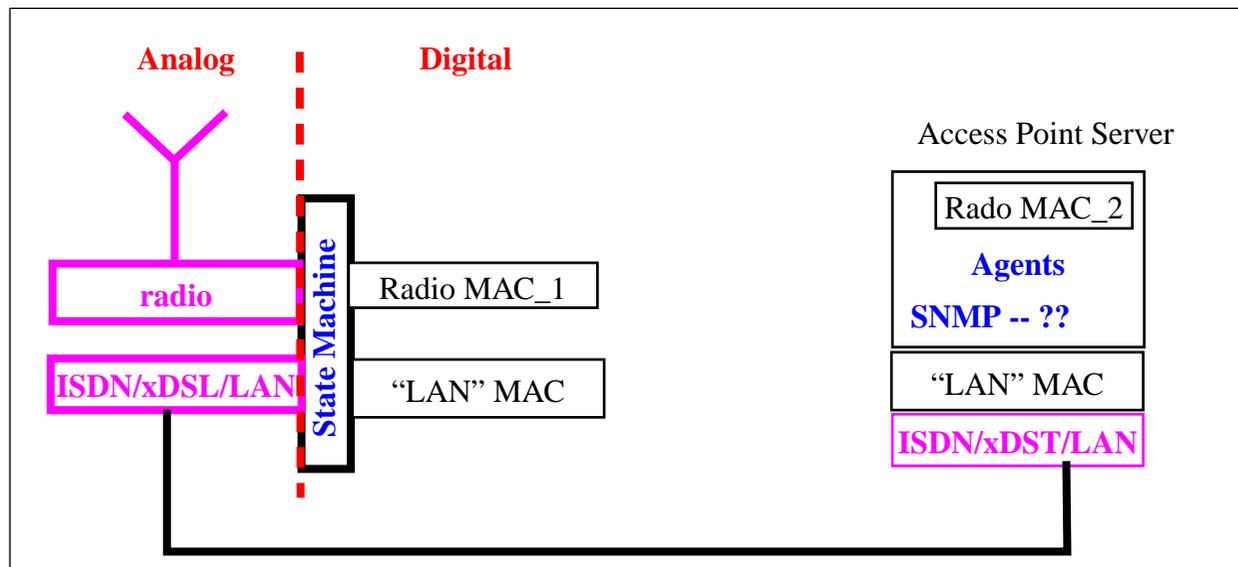# MEDIA

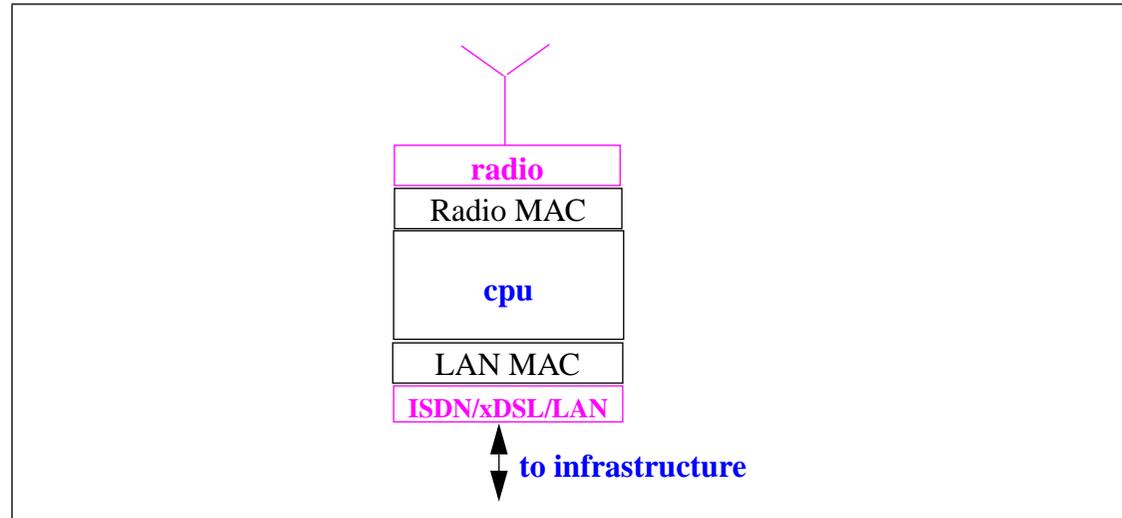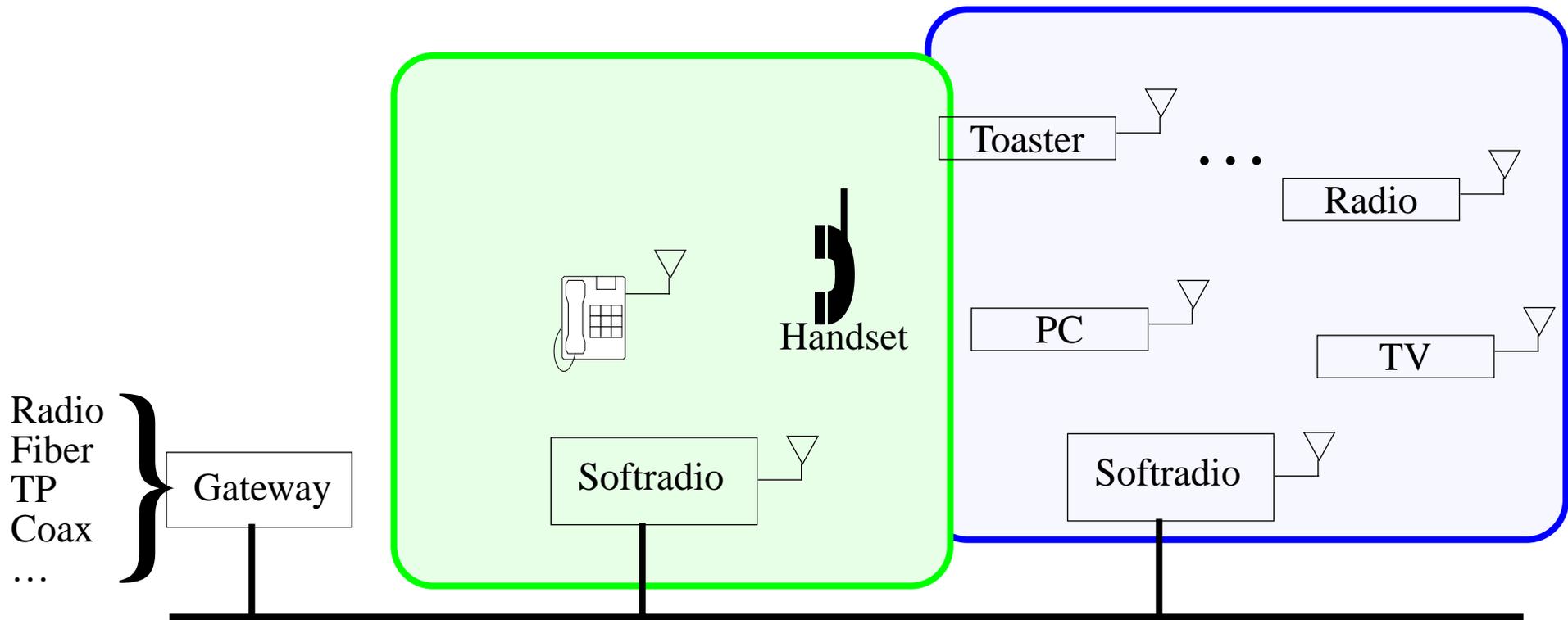High integration (goal of MEDIA project)
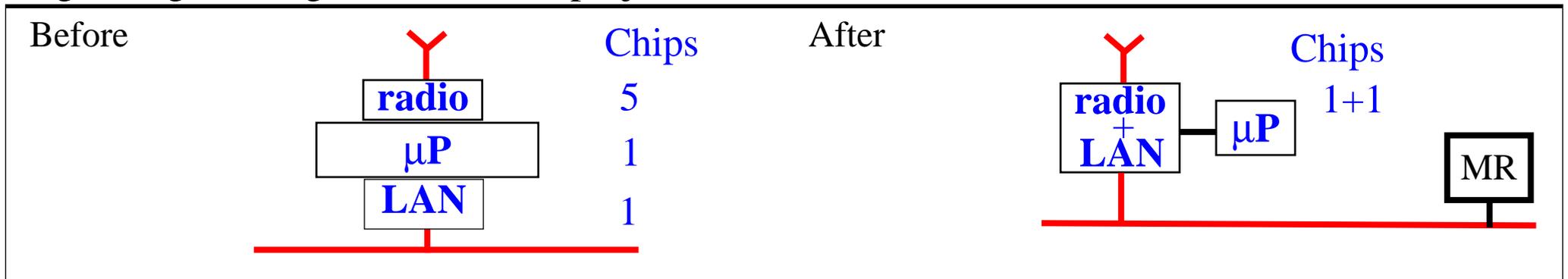
Partners:

- Kungl Tekniska Högskolan (KTH/ELE/ESDlab and KTH/IT/CCSlab)

- Tampere University of Technology (TUT)

- GMD FOKUS (GMD)

- Technische Universität Braunschweig (UBR)

- Interuniversity Microelectronics Centre (IMEC)

- Ericsson Radio Systems AB (ERA)

See http://www.ele.kth.se/ESD/MEDIA for more information

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 729 of 740
Internetworking/Internetteknik

Split the functions between access point and access point server

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 730 of 740
Internetworking/Internetteknik

Future home/office/… network accesspoints

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 731 of 740
Internetworking/Internetteknik

Personal Computing and Communication (PCC)

Upper limit of bandwidth: saturate the senses: sight, sound, touch, smell, taste

$$\Rightarrow \sim 1 \text{ Gbit/sec/user}$$

Current workstations shipping with 1 Gbit/sec interfaces for LAN!

Telepresense for work is the long-term "killer" application

-- Gordon Bell and James N. Gray[1]

---

1. "The Revolution Yet to Happen" in Beyond Calculation: The Next Fifty Years of Computing, Eds. Denning and Metcalfe, Copernicus, 1997.

Maguire
maguire@it.kth.se
Future networks
2006.03.13
Future and Summary 732 of 740
Internetworking/Internetteknik

Uploading ourselves to the net

In Bob Metcalf's speech at MIT: http://web.mit.edu/alum/president/speech.html

One of great insights of this talk is that the internet is the way to immortality[1]:

> Now, for the next 50 years, the web will drive electronic commerce into the information age, ubiquitous computers will disappear into the woodwork, and we'll start uploading ourselves into the Internet to become at last immortal.
>
> -- Robert M. Metcalfe
> June 26, 1997

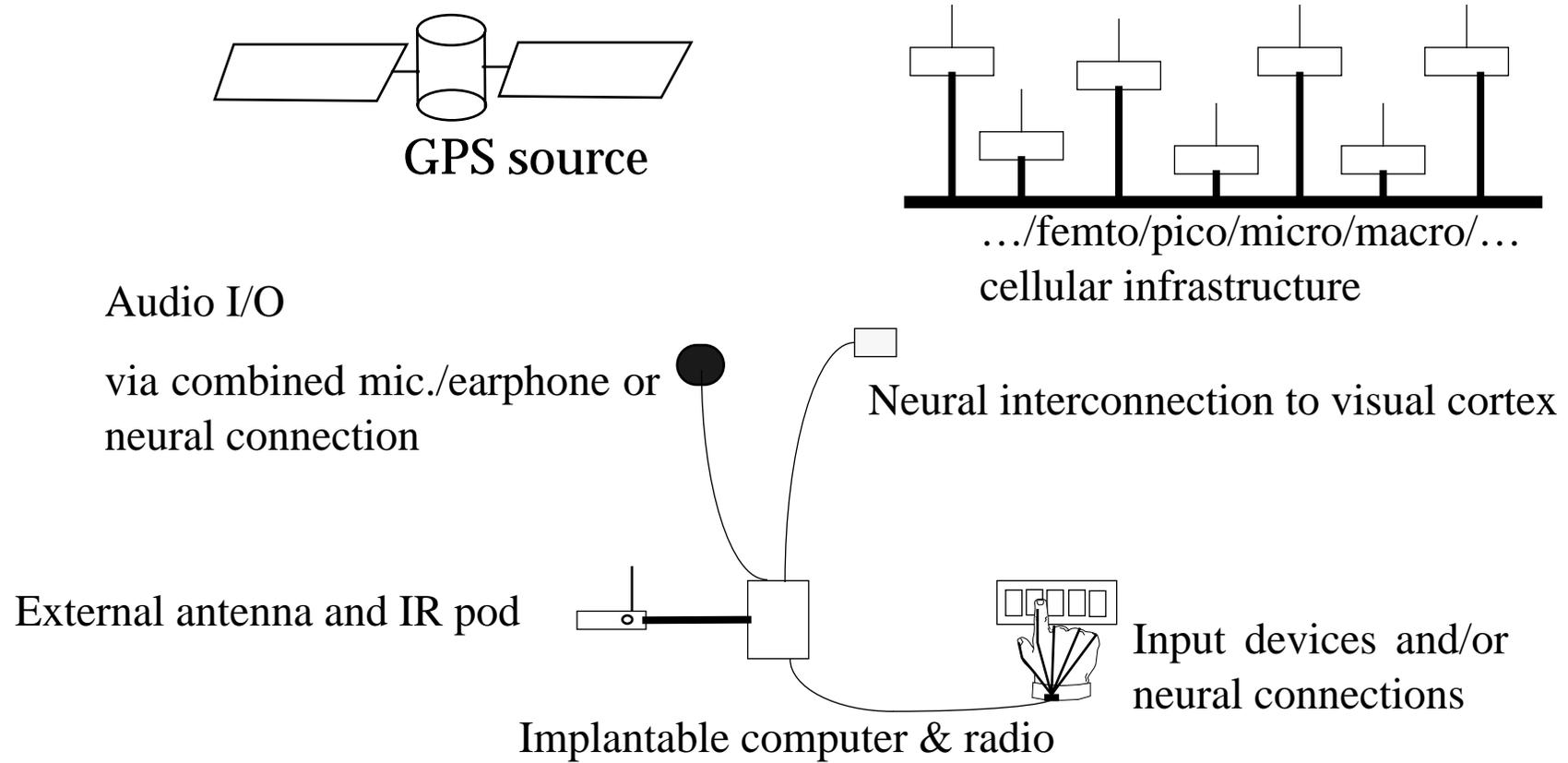1. Robert M. Metcalfe, "Internet Futures", MIT Enterprise Forum, June 26, 1997.

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 733 of 740
Internetworking/Internetteknik

Future Systems

GPS source

…/femto/pico/micro/macro/…
cellular infrastructure

Audio I/O

via combined mic./earphone or
neural connection

Neural interconnection to visual cortex

External antenna and IR pod

Input devices and/or
neural connections

Implantable computer & radio

Figure 124: Vision-3, 2005-2015 - very high level of integration

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 734 of 740
Internetworking/Internetteknik

Bionic Technologies, Inc.'s Intracortical Electrode Array

Acute microelectrode assembly (10x10 array, 100 active electrodes) $1,250.00
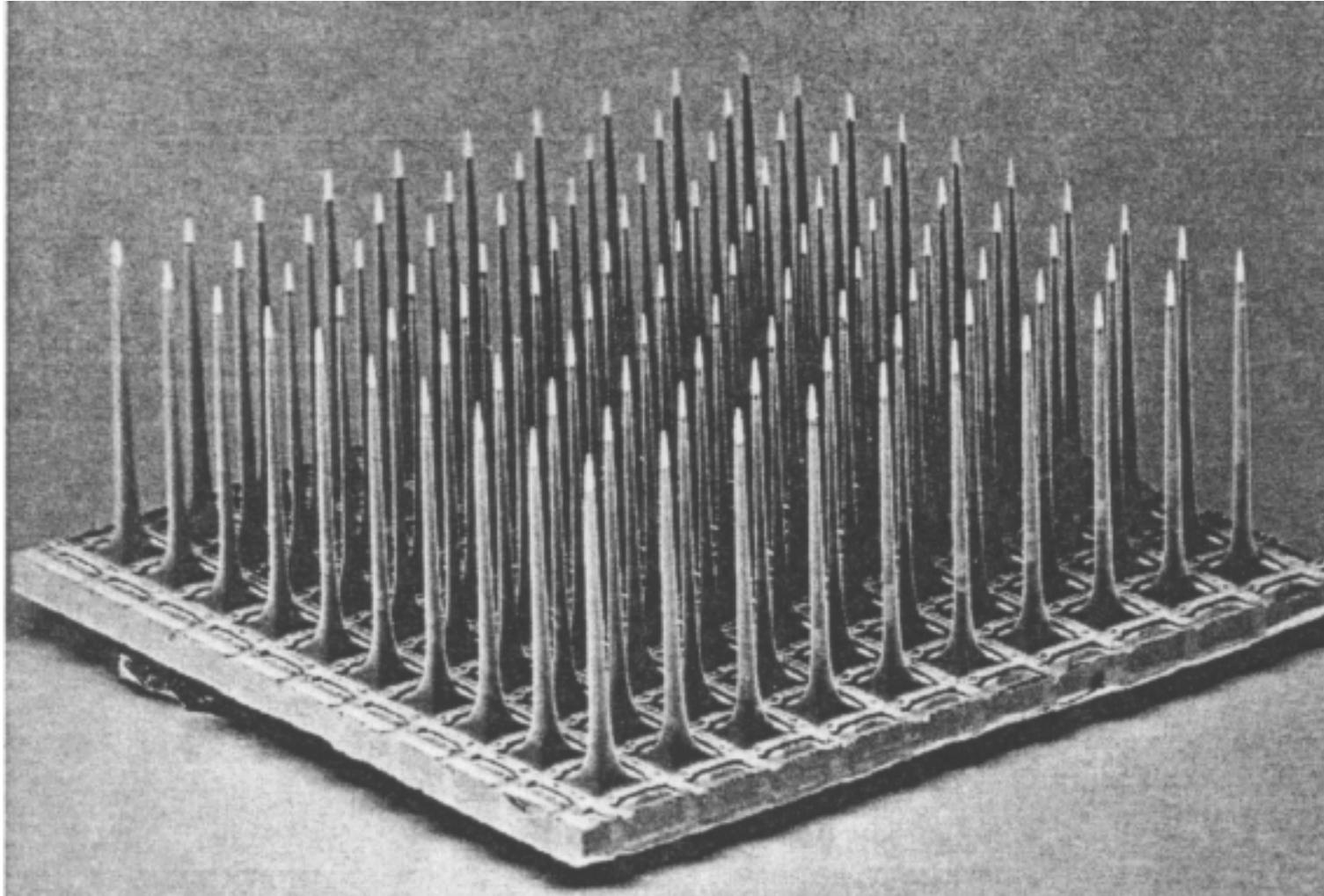

Figure 125: 10 x 10 silicon electrode array (each electrode: 1.5mm long, 0.08mm wide at base, 0.001mm tip), Built at the Univ. of Utah, by Richard A. Normann, et al.; from Scientific American, March 1994, pg. 108.

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 735 of 740
Internetworking/Internetteknik

Non-metalic bi-directional neural interfaces

Neurochip: Neuron silicon circuits   http://mnphys.biochem.mpg.de/ :

(a) Silicon-Neuron Junction (input to the nerve)   (b) Neuron transistor (output from the nerve)
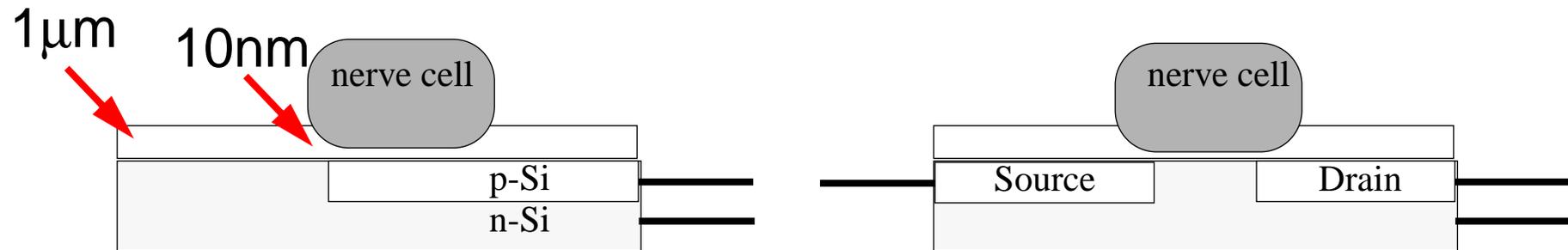

Figure 126: (a) Capacitive coupling of data into nerve and
(b) using the charge in the nerve to control a transistor's gate for getting data out of the nerve

(a) Peter Fromherz and Alfred Stett, "Silicon-Neuron Junction: Capacitive Stimulation of an Individual Neuron on a Silicon Chip" Phys.Rev.Lett. 75 (1995) 1670-1673

(b) P.Fromherz, A.Offenhäusser, T.Vetter, J.Weis, "A Neuron-Silicon Junction: A Retzius-Cell of the Leech on an Insulated-Gate Field-Effect Transistor" Science 252 (1991) 1290-1293.

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 736 of 740
Internetworking/Internetteknik

What is your time line?

- What is going to be your planning horizon?

- What will be the depreciation time for your equipment/software/infrastructure/… ?

- How fast:

    ♦ can you change?

    ♦ should you change?

    ♦ will you change?

Maguire
maguire@it.kth.se

Future networks
2006.03.13

Future and Summary 737 of 740
Internetworking/Internetteknik

Summary

- Telecom operators are reinventing themselves and their infrastructures
- Things to watch IPv6, IPsec, Mobile-IP, DHCP, the new domain name registries, appliances, …
- Low cost access points which exploit existing or easily installed infrastructure are key to creating a ubiquitous mobile infrastructure with effectively infinite bandwidth.
- Smart Badge is a vehicle for exploring our ideas:
    - Exploits hardware and software complexity by hiding it.
    - Explores allowing devices and services to use each other in an extemporaneous way.
    - Enables a large number of location and environment aware applications, most of which are service consuming.
    - Keep you eyes open for the increasing numbers of senors which will be on the network.
    - Service is where the money is!
- Personal Communication and Computation in the early 21st century: "Just Wear IT!"
- Coming in 20-30 years: "Just implant IT!"
- Remember: The internet will be what you make it.

Maguire
maguire@it.kth.se

Summary
2006.03.13

Future and Summary 738 of 740
Internetworking/Internetteknik

Further Reading

[139]Kalevi Kilkki, Differentiated Services for the Internet, Macmillan Technical Publishing, 384 pages, June 1999, ISBN: 1578701325.

[140] PCI-SIG, PCI-X 2.0: High Performance, Backward Compatible PCI for the Future, May 19, 2005 `http://www.pcisig.com/specifications/pcix_20`

[141] USB.org, Universal Serial Bus Revision 2.0 specification, May 19, 2005

`http://www.usb.org/developers/docs/usb_20_02212005.zip`

[142]Tom Clark, *IP SANS: A Guide to iSCSI, iFCP, and TCIP Protocols for Storage Area Networks*, Addison-Wesley, 288 pages, 2002, ISBN: 0-201-75277-8

General: `http://www.ietf.org/`

Maguire
maguire@it.kth.se

Further Reading
2006.03.13

Future and Summary 739 of 740
Internetworking/Internetteknik

Thanks

Best wishes on your written assignments (or projects).

Maguire

maguire@it.kth.se

Thanks

2006.03.13

Future and Summary 740 of 740

Internetworking/Internetteknik