# TCP/IP Routing

Alan Altmark
IBM Corporation
Endicott, New York

RETURN TO INDEX

This presentation provides in-depth information on configuration of the routing components of VM TCP/IP FL320.

References to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe on any of the intellectual property rights of IBM may be used instead. The evaluation and verification of operation in conjunction with other products, except those expressly designed by IBM, are the responsibility of the user.

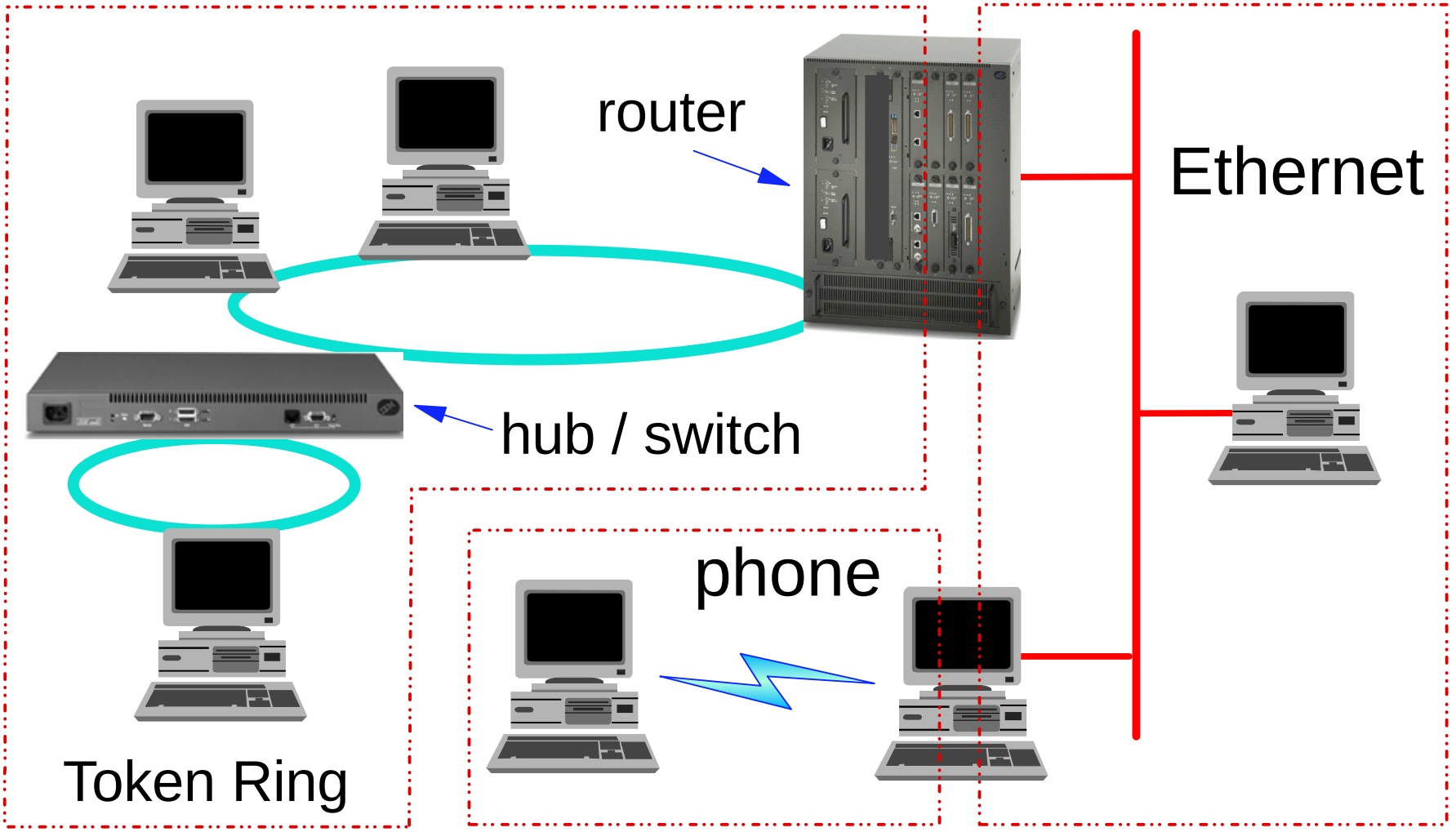
The following terms are trademarks of the IBM Corporation in the United States or other countries or both: S/390 VM/ESA IBM OS/390

Other company, product, and service names, which may be denoted by double asterisks (**), may be trademarks or service marks of others.

# Agenda

- IP Addressing
  - Classes
  - Subnetting
- Link-level communications
  - MAC frames
  - ARP
- Routing basics
- Virtual IP addressing

# Terminology: LAN Segment



# IPv4 Addressing

- 32-bit address, 4 octets
  - High-order bits identify network
  - Low-order bits identify host within network
  - Expressed as **a.b.c.d**
- Special values for network and host
  - All ones = "everyone"
  - All zeros = "me", "this", or "default"
- Address space divided into classes

# IPv4 Addressing: Class A



- Networks: 0 to 127
  Total: 128 networks

**9.130.57.21**

| 9 | 130 | 57 | 21 |
|---|---|---|---|
| 0x09 | 0x82 | 0x39 | 0x15 |
| 0000 1001 | 1000 0010 | 0011 1001 | 0001 0101 |

Network 9 | Host 8 534 293

# IPv4 Addressing: Class B



- Networks: 128.0 to 191.255
  Total: 16 384 networks

**148.100.204.3**

| 148 | 100 | 204 | 3 |
|---|---|---|---|
| 0x94 | 0x64 | CCx | 0x03 |
| 1001 0100 | 0110 0100 | 1100 1100 | 0000 0011 |
| Network 148.100 | | Host 52 227 | |

# IPv4 Addressing: Class C



- Networks: 192.0.0 to 223.255.255
  Total: 2 097 152 networks

**200.14.64.191**

| 200 | 14 | 64 | 191 |
|---|---|---|---|
| 0xC8 | 0x0E | 0x40 | 0xBF |
| 1100 1000 | 0000 1110 | 0100 0000 | 1011 1111 |
| Network 200.14.64 | | | Host 191 |

# IPv4 Addressing: Classes D & E



- Class D
  - 224.0.0.0 to 239.255.255.255
  - high-order bits = 1110
  - provides 28-bit multicast group id
- Class E
  - 240.0.0.0 to 247.255.255.255
  - high-order bits = 11110
  - Not used

# Subnetting

- Class A and B networks provide for 16M and 64K hosts, respectively
- LAN segments do not contain anywhere near that many hosts
- Divide up host id portion of address into managable groups called subnets
  - Can subnet class C networks, too

# Subnetting

- Hosts that are members of the same subnet are considered to be in the same LAN segment
  - ATM hosts may be on separate physical LAN segments
  - Point-to-point
- Multiple subnets may share same LAN segment

# Subnetting

- The class mask defines which bits of the host id are used for the subnet number

- Subnet = bitand(address,  mask)

  Perform logical AND of destination address and subnet mask to get subnet number

# IPv4 Subnet Addressing

**Subnet mask = 255.255.255.0 (24 bits)**

**IP address = 9.130.57.21**

| 9 | 130 | 57 | 21 |
|---|---|---|---|
| 0x09 | 0x82 | 0x39 | 0x15 |
| 0000 1001 | 1000 0010 | 0011 1001 | 0001 0101 |
| Network | Subnet | | Host |

**Network = 9**

**Subnet = 9.130.57**

**Host = 21**

# IPv4 Subnet Addressing

**Subnet mask = 255.255.255.192 (26 bits)**

**IP address = 9.130.1.181**

| | | | |
|---|---|---|---|
| 9 | 130 | 1 | 181 |
| 0x09 | 0x82 | 0x01 | 0xB5 |
| 0000 1001 | 1000 0010 | 0000 0001 | 10 11 0101 |
| Network | Subnet | | Host |

**Network = 9**

**Subnet = 9.130.1.128 (say, what?)**

**Host = 53 (eh?)**

# IPv4 Subnet Addressing



**Subnet mask = 255.255.255.192 (26 bits)**

**IP address = 9.130.1.181**

| | | | | |
|---|---|---|---|---|
| | 0000 1001 | 1000 0010 | 0000 0001 | 1011 0101 |
| & | 1111 1111 | 1111 1111 | 1111 1111 | 1100 0000 |
| = | 0000 1001 | 1000 0010 | 0000 0001 | 1000 0000 |
| = | 9 | 130 | 1 | 128 |

**Subnet = 9.130.1.128**

**Host = 53 (0x35)**

| |
|---|
| 0011 0101 |

Remaining bits are host number

# IP Addressing Cheat Sheet



| class | first octet | network |
|---|---|---|
| A | 0-127 | a.0.0.0 |
| B | 128-191 | a.b.0.0 |
| C | 192-223 | a.b.c.0 |
| D | 224-239 | n/a |

| mask size | last octet | binary | subnetworks | hosts |
|---|---|---|---|---|
| /25 | 128 | 1000 0000 | 2: 0 128 | 126 |
| /26 | 192 | 1100 0000 | 4: 0 64 128 192 | 62 |
| /27 | 224 | 1110 0000 | 8: 0 32 64 96 128 160 192 224 | 30 |
| /28 | 240 | 1111 0000 | 16: 0 16 32 48 64 80 96 112 ... | 14 |
| /29 | 248 | 1111 1000 | 32: 0 8 16 24 32 40 48 56 64 ... | 6 |
| /30 | 252 | 1111 1100 | 64: 0 4 8 16 20 24 28 32 36 ... | 2 |

# Special IPv4 Addresses

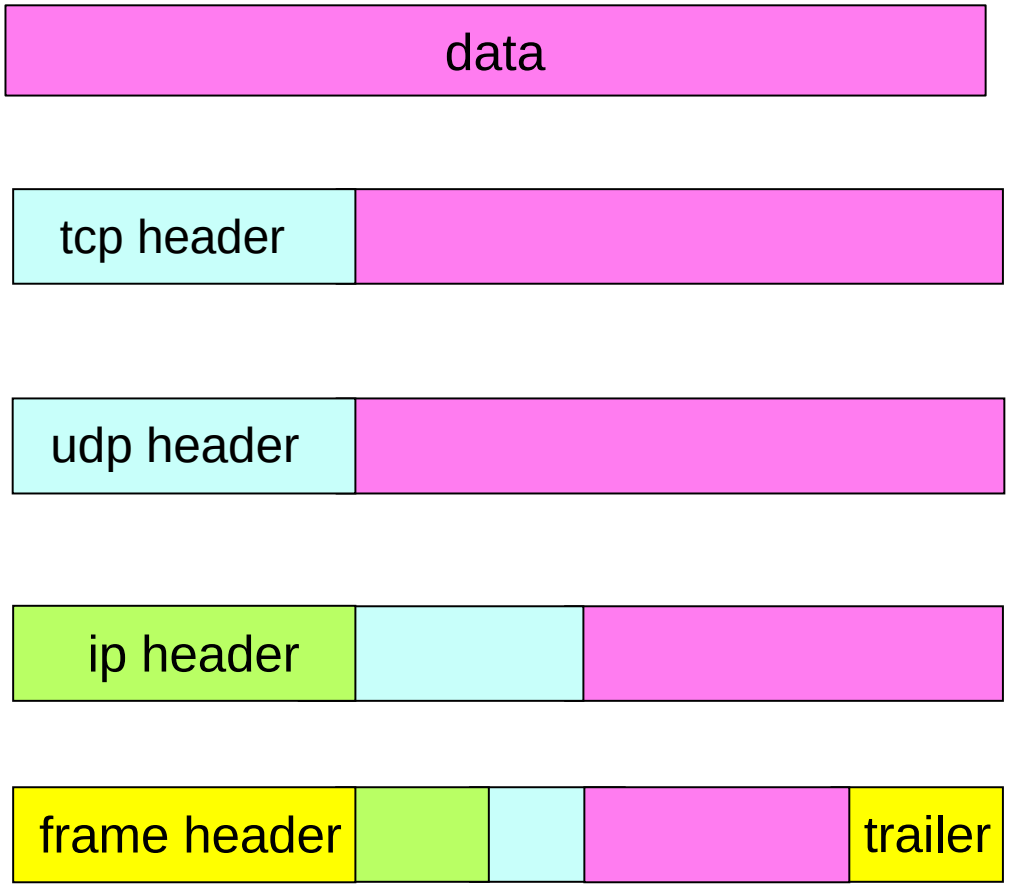
| net ID | subnet ID | host ID | Source | Destination | Description |
|---|---|---|---|---|---|
| 0 | | 0 | yes | no | this host on this net |
| 0 | | *hostid* | yes | no | specific host on this net |
| 127 | | *any* | yes | yes | Loopback |
| -1 | | -1 | no | yes | local media broadcast |
| *netid* | | -1 | no | yes | network-directed broadcast |
| *netid* | *subnetid* | -1 | no | yes | subnet-directed broadcast |
| *netid* | -1 | -1 | no | ok | all-subnets-directed broadcast |

Local broadcasts are not bridged or routed to other LAN segments

# Basic Communications:
## Terminology



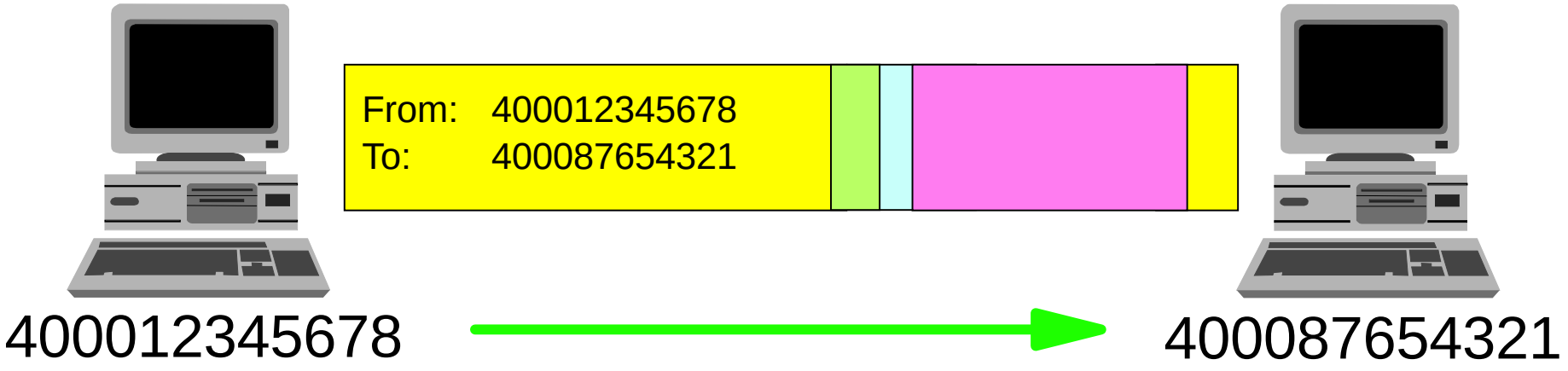
- Application data
- TCP Segment
- UDP Datagram
- IP Datagram
- Link Frame



# Basic Communications:
## Link Addressing



- Frames transmitted using Medium Access Control points and addresses

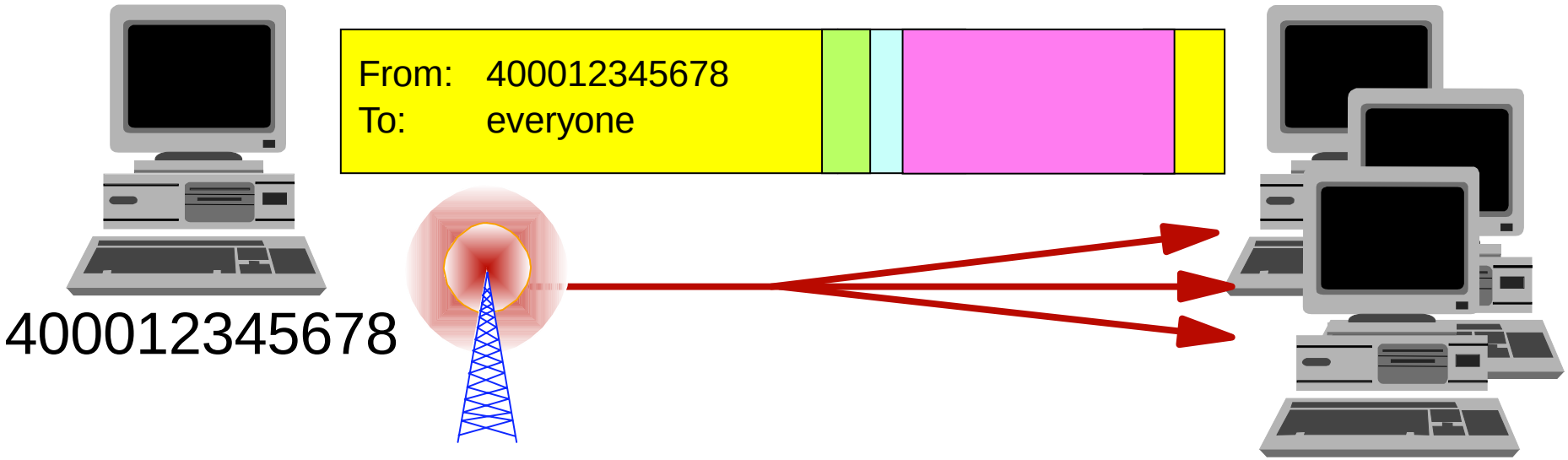



- Maximum Transmission Unit (MTU) limits message length
  - May force IP datagram fragmentation

# Basic Communications:
## Link Broadcast



- Station can broadcast by using a special destination MAC address




- All stations will pick up the frame

# Basic Communications:
## Physical vs. Logical Addressing

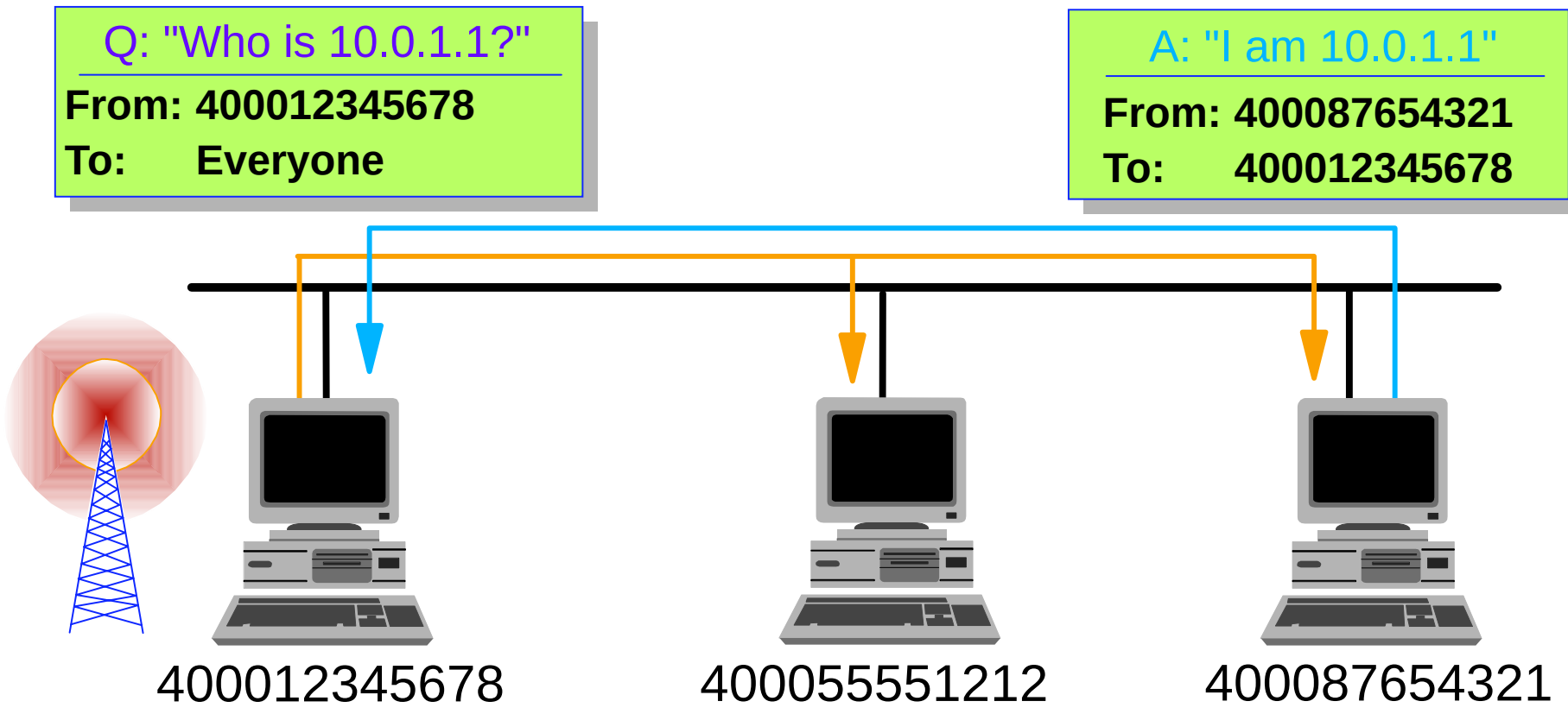


- Problem:
  - TCP/IP hosts are configured to use logical IP addresses, not physical MAC addresses
  - How does TCP/IP find out what MAC address to send data to?
- Answer:
  Address Resolution Protocol (ARP)

# Basic Communications:
## Address Resolution Protocol



- Address Resolution Protocol, ARP, allows host to determine MAC address



# Basic Communications:
## Address Resolution Protocol



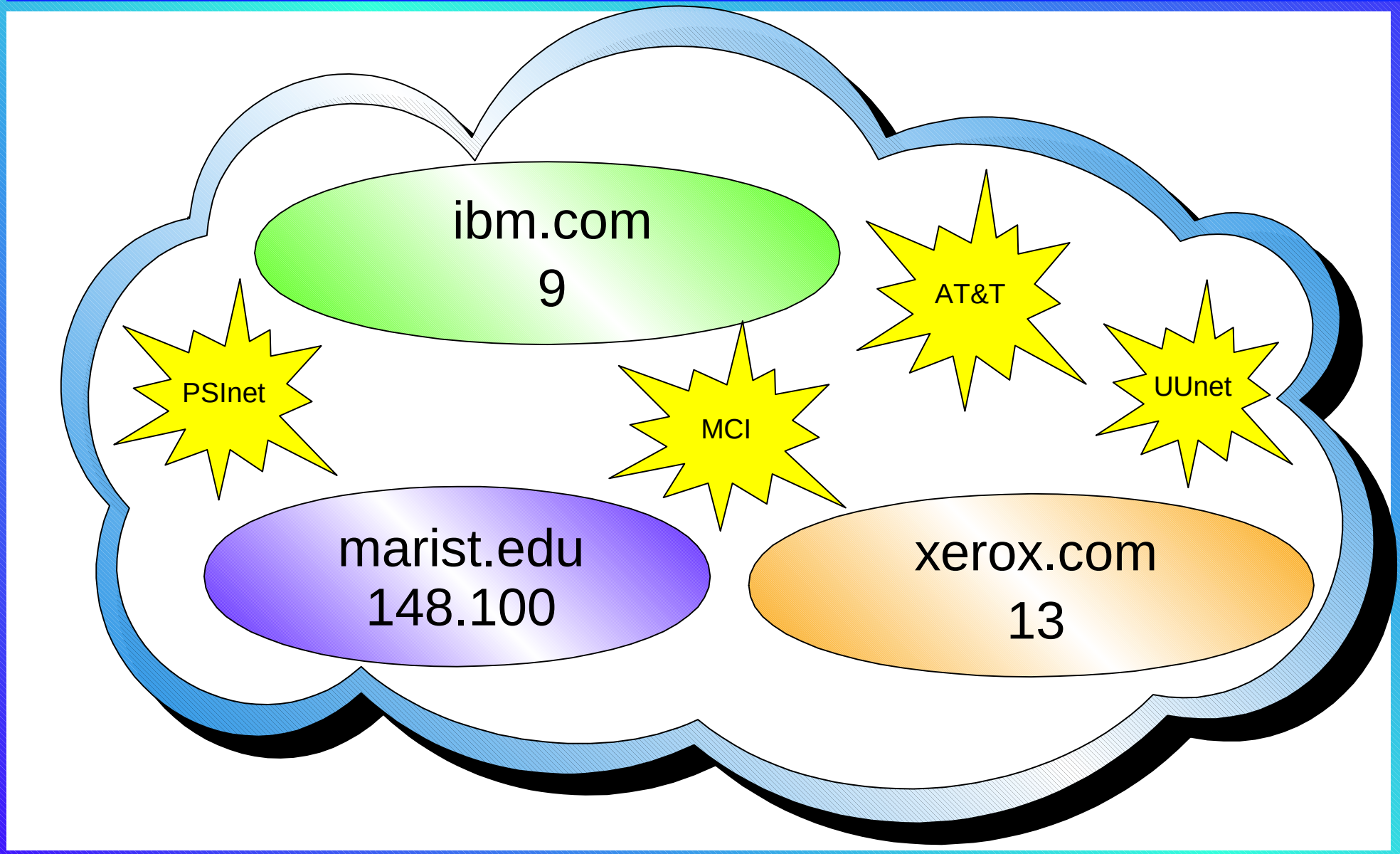
- Hosts maintain a cache of ARP responses to avoid ARP before sending each frame
- ARP cache entries expire so that hosts can discover MAC address changes
  - New adapter
  - Different box with same IP address
    e.g. hot standby
  - PROFILE TCPIP

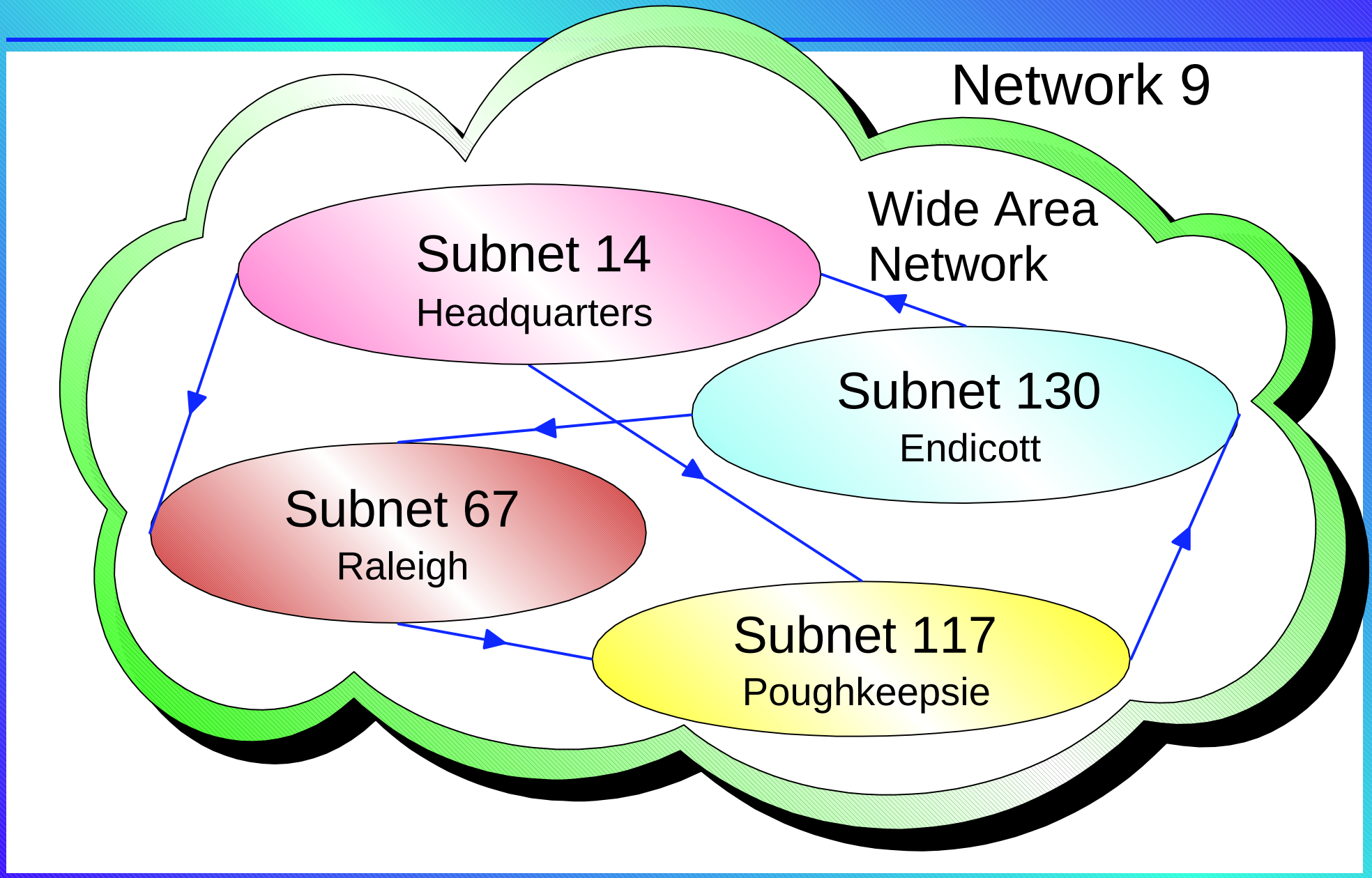
# Basic Communications:
## Address Resolution Protocol

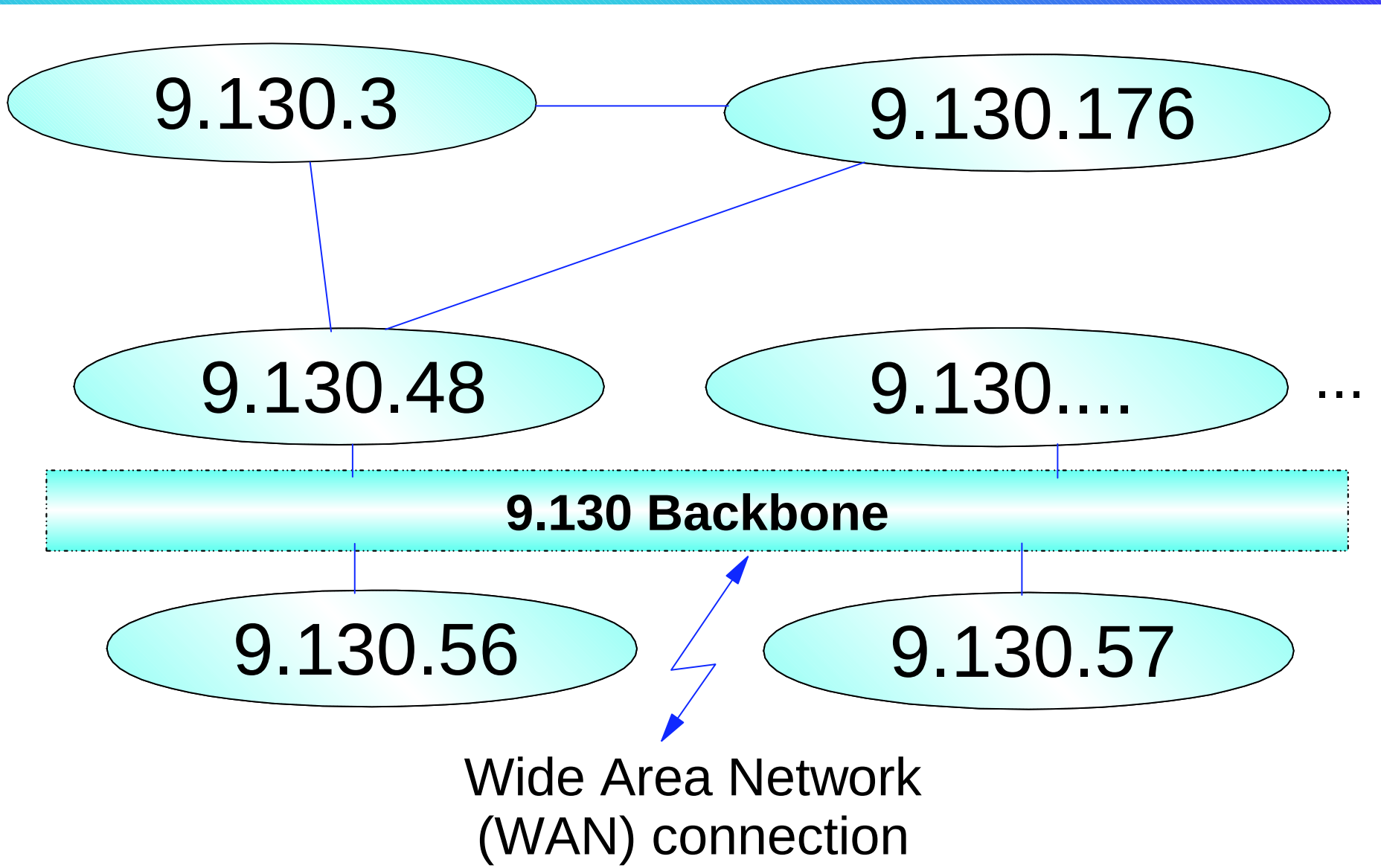


- Local hosts are on same LAN segment and can be reached via ARP
- Remote hosts must be reached through a local gateway or router
  - Each host has a default gateway defined to it

# Networks on the Internet



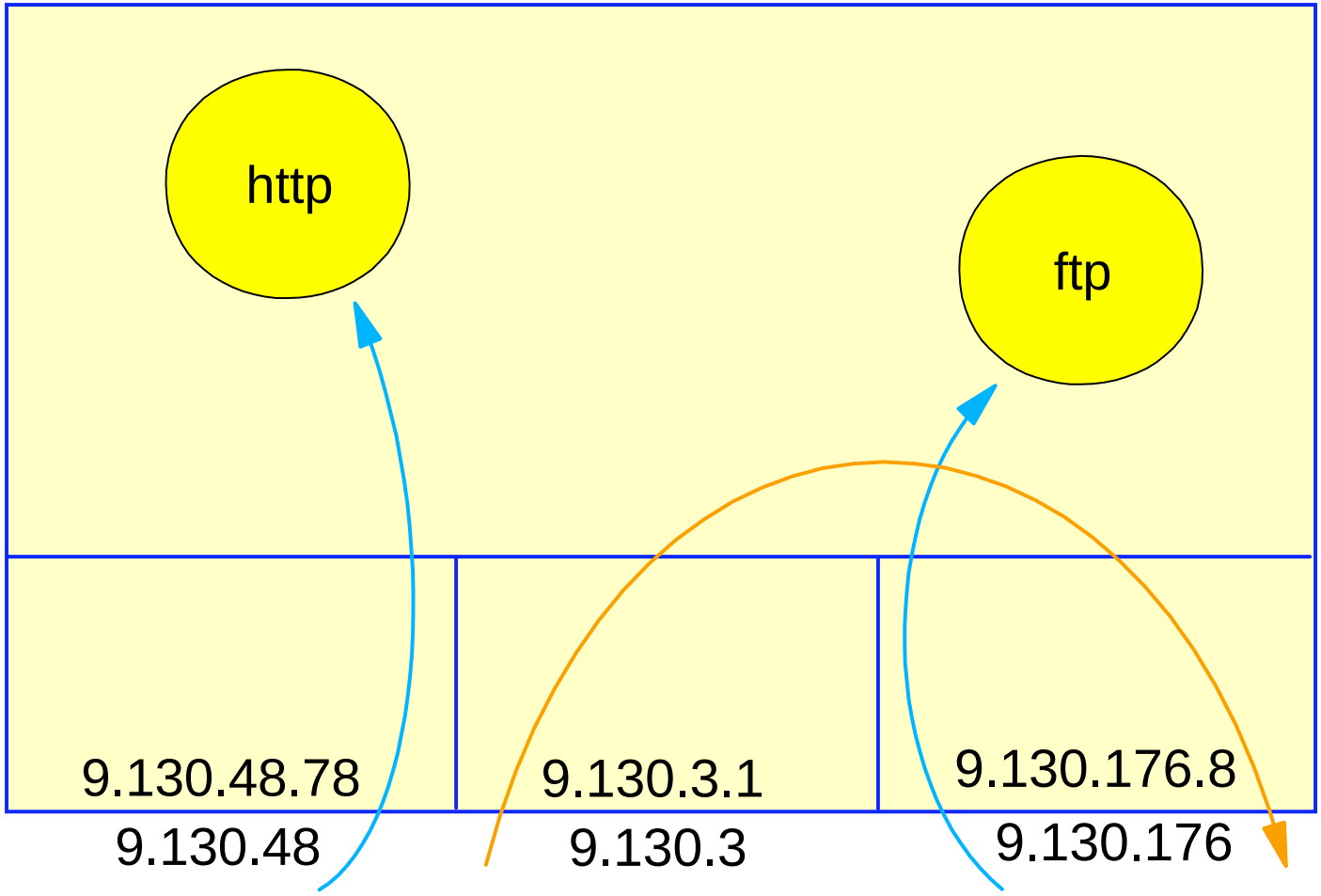

Network 9
Wide Area Network
Subnet 14
Headquarters
Subnet 130
Endicott
Subnet 67
Raleigh
Subnet 117
Poughkeepsie


9.130.3
9.130.176
9.130.48
9.130....
...
9.130 Backbone
9.130.56
9.130.57
Wide Area Network
(WAN) connection

# IP Packet Routing

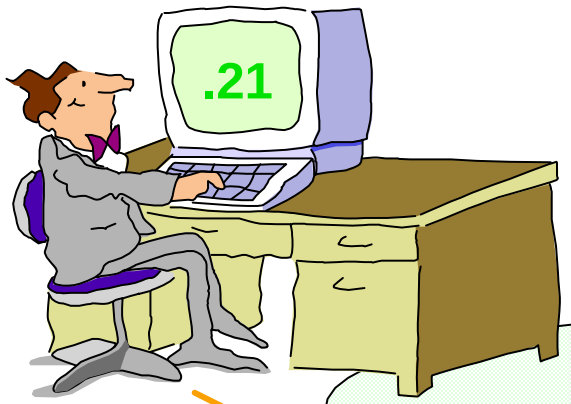
- Occurs whenever an IP packet is received or sent by a host
- Sometimes trivial
  - Only one possible route
- Sometimes complex
  - Multi-homed host

# Multi-Homed Host




- MTU may be different on each interface!

# IP Packet Routing




default gateway
= .253

.253

9.130.57

.253

9.130.48

*Message*
From: 9.130.57.21
To: 9.130.3.4
Via: 9.130.57.253

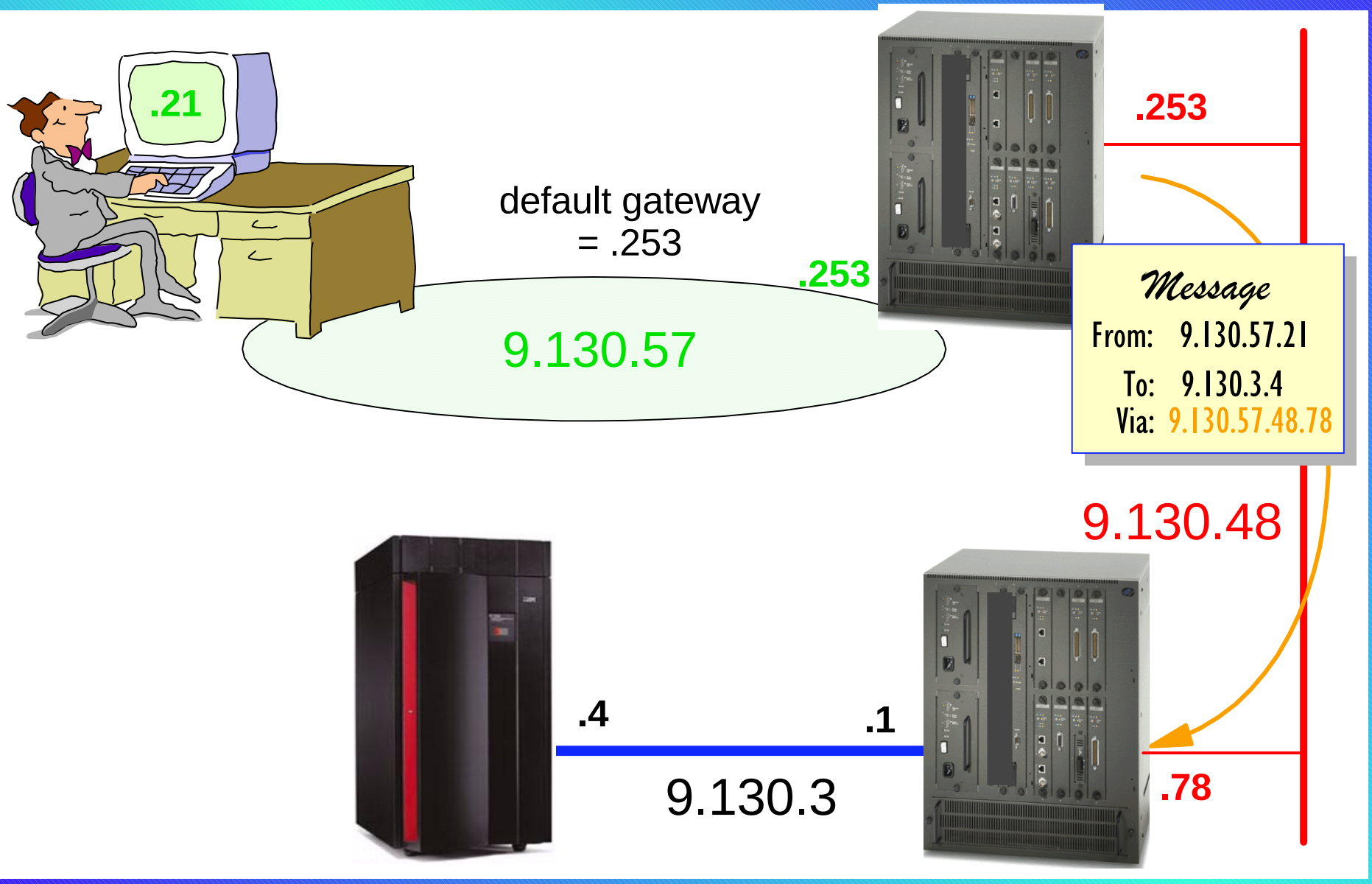
.4

.1

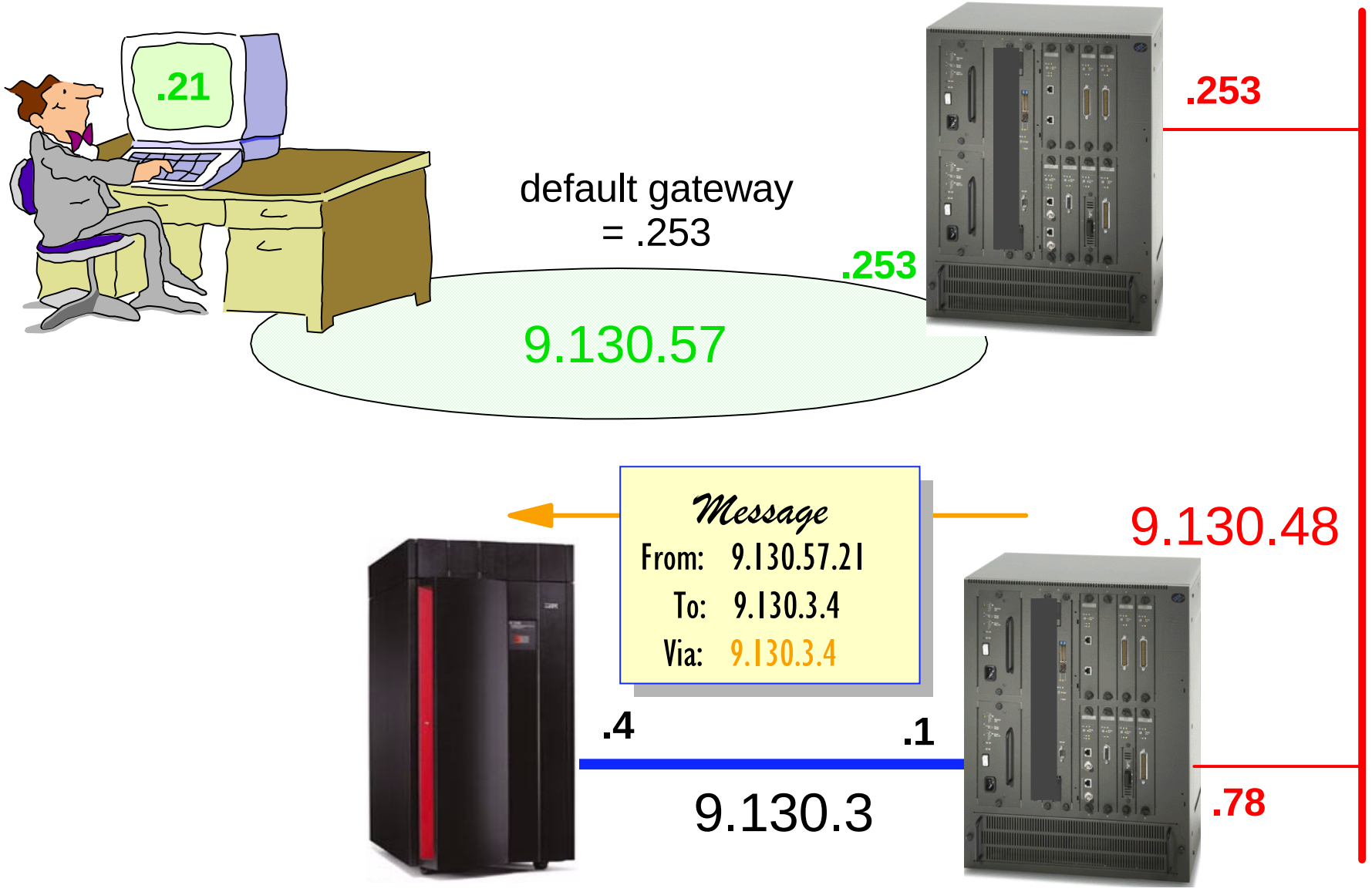
9.130.3

.78

# IP Packet Routing



# IP Packet Routing



# Routing Configuration

9.130.48.253

B 128.84

128.84.176 / 24

9.130.48.78 A

C 9.130.3.1

9.130.48 / 24

9.130.3.4

internet

1. What networks are directly connected?
2. Who handles indirect destinations?

# Static Routing Definition



| Destination | Via | Subnet mask | Subnet value | Link name |
|---|---|---|---|---|
| 9.0.0.0 | direct | 255.255.255.0 | 9.130.48 | A |
| 128.84.0.0 | direct | 255.255.255.0 | 128.84.176 | B |
| 9.130.3.4 | direct | n/a | n/a | C |
| default | 9.130.48.253 | | | |

**Home**

```
   9.130.48.78     A
   128.84.176.8    B
   9.130.3.1       C
```

## Use Gateway - not BsdRoutingParms

**Network** **first hop** **link mtu** **subnet mask** **subnet value**

**Gateway**

```
  9              =          A  2000  0.255.255.0  0.130.48.0
  128.84         =          B  1500  0.0.255.0    0.0.176.0
  9.130.3.4      =          C  4000  HOST
  defaultnet     9.130.48.253 A  2000  0
```

# GATEWAY Arcana

- Network
  - Value is network only, not subnet
  - Network value depends on class
  - Trailing zeros may be omitted
  - Must provide a default, **defaultnet**
- First hop
  - "=" indicates direct link, or
  - Must be host for which a route exists

# GATEWAY Arcana

- Subnet Mask
  - network bits must be zero
  - zero indicates no subnetting
  - "HOST" indicates point-to-point link
    - Sometimes seen as mask 255.255.255.255
- Subnet Value
  - zero indicates default route for subnet
  - bits not defined by mask must be zero

# Dynamic Routing

- RouteD servers communicate routing information
  - Routing Information Protocol, RIP
  - Status of local links (up / down)
  - List of directly connected networks
  - Routes to other networks or hosts learned from other servers
- Modifies IP routing table in stack
  - Provides network topology

# Routing Information Protocol

- Broadcasts everything it knows every 30 seconds
- Listens for broadcasts from other routed servers
  - At least two servers required!
  - Must be reminded every 3 minutes
- Broadcasts link up / down immediately

# Dynamic Routing Definition



| Destination | Via | Subnet mask | Subnet value | Link name |
|---|---|---|---|---|
| 9.0.0.0 | direct | 255.255.255.0 | 9.130.48 | A |
| 128.84.0.0 | direct | 255.255.255.0 | 128.84.176 | B |
| 9.130.3.4 | direct | n/a | n/a | C |

```
Home
    9.130.48.78     A
    9.130.176.8     B
    9.130.3.1       C
```

## Use BsdRoutingParms - not Gateway

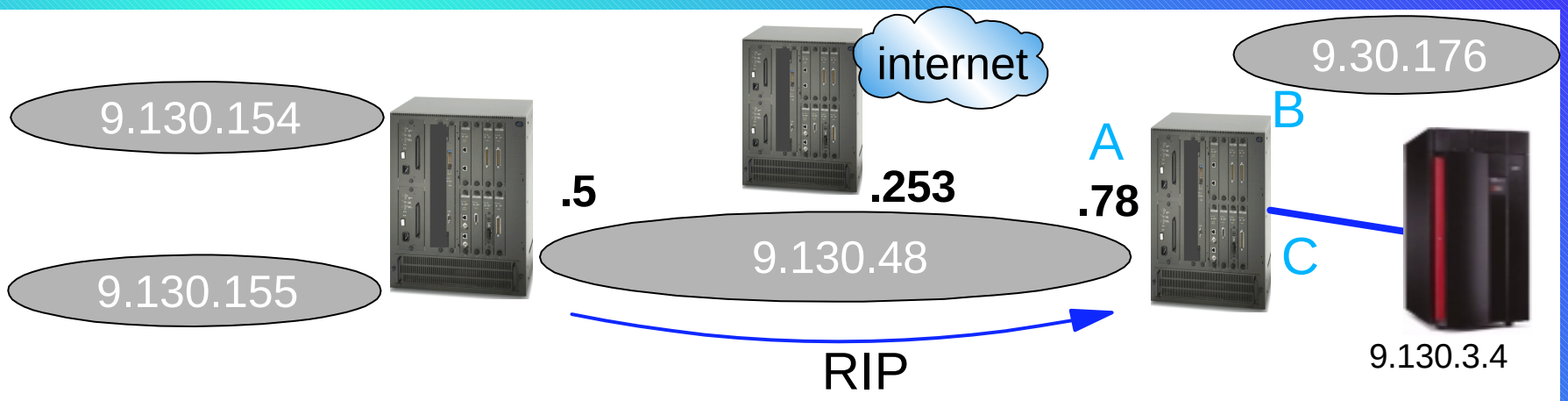
```
link   mtu    metric   subnet mask       destination address
BsdRoutingParms false
  A      2000    0      255.255.255.0           0
  B      1500    0      255.255.255.0           0
  C      4096    0      255.255.255.0      9.130.3.4
EndBsdRoutingParms
```

# BSDroutingParms Arcana



- Not the same as Gateway statement!
  - network is obtained from Home statement
  - subnet mask **includes** network bits
  - default is **learned**, not pre-defined
- Metric defines number of hops to destination
  - Lower values are preferred routes
  - Used by other routers
  - Gives you a way to persuade packets to move in a particular direction

# Routing table after 1 minute




| Destination | Via | Subnet mask | Subnet value | Link name |
|---|---|---|---|---|
| 9.0.0.0 | direct | 0.255.255.0 | 0.130.48.0 | A |
| 9.0.0.0 | direct | 0.255.255.0 | 0.130.176.0 | B |
| 9.130.3.4 | direct | host | | C |
| **default** | **9.130.48.253** | | | **A** |
| **9.0.0.0** | **9.130.48.5** | **0.255.255.0** | **0.130.154.0** | **A** |
| **9.0.0.0** | **9.130.48.5** | **0.255.255.0** | **0.130.155.0** | **A** |

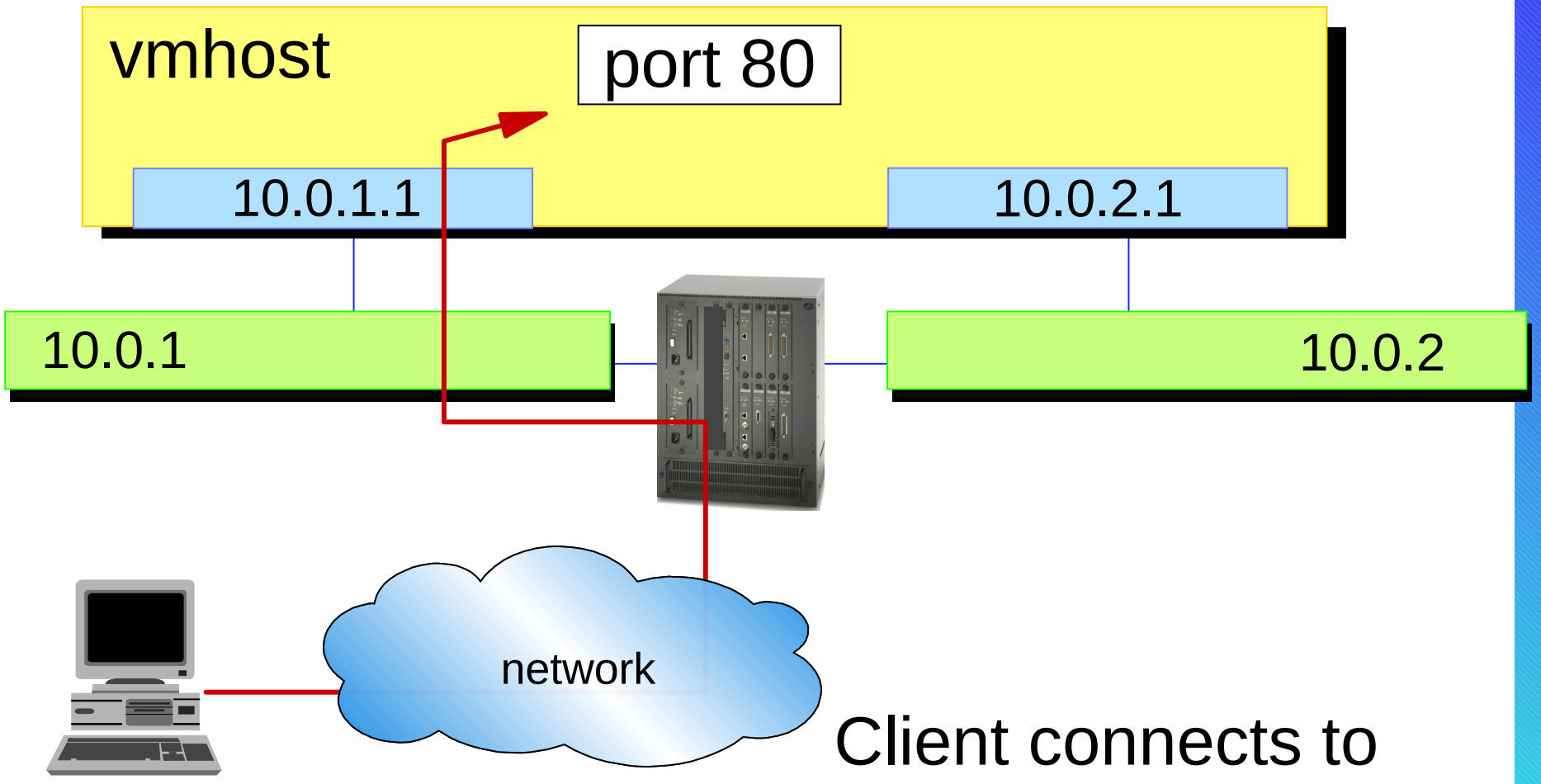
- If .5 stops broadcasting, .78 will forget all routes sent by it

# NETSTAT GATE



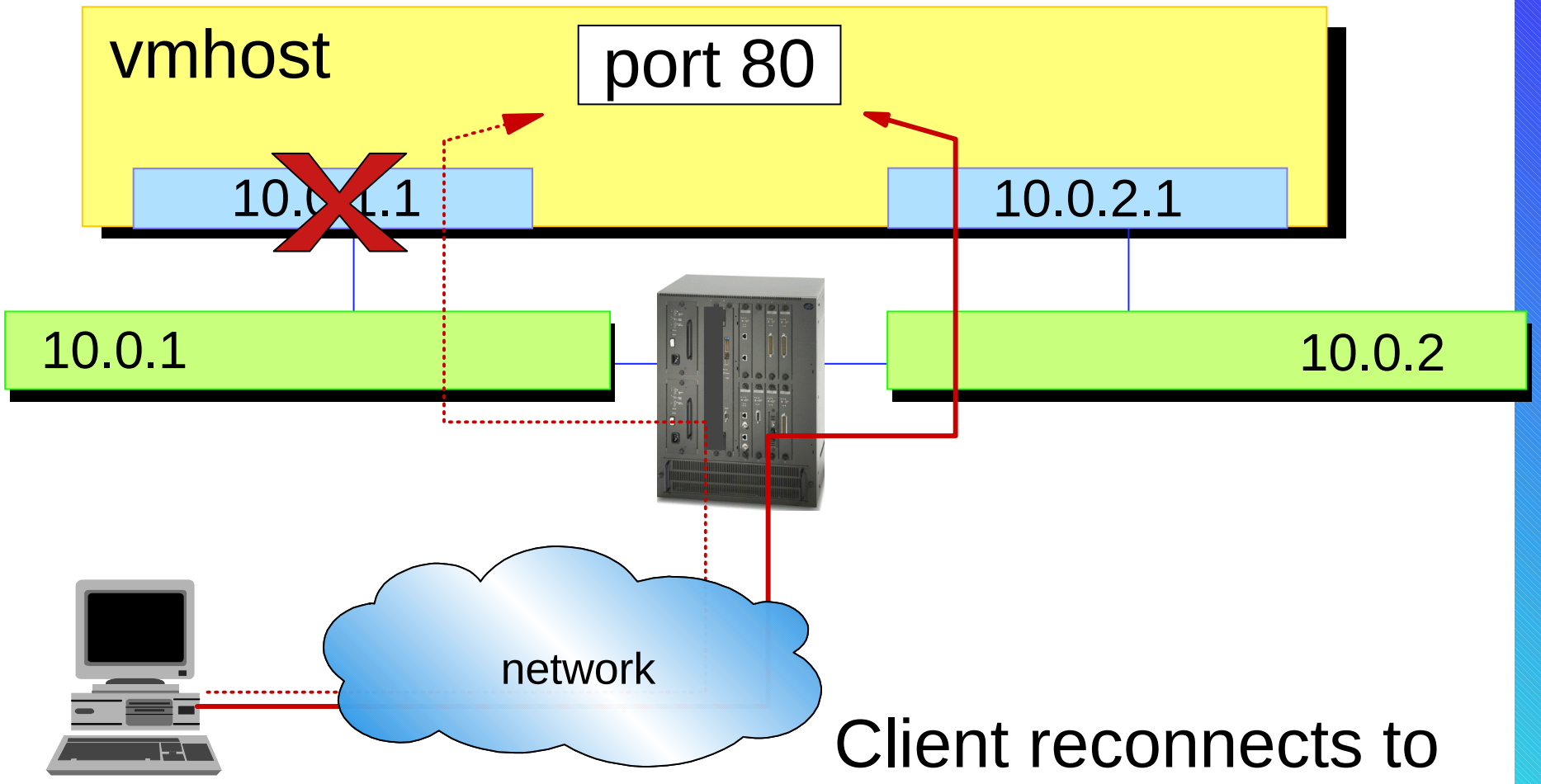
```
NetAddress      FirstHop        Link     Pkt Sz  Subnet Mask     Subnet Value
----------      --------        ------   ------  -----------     ------------
Default         9.130.48.253    ISRING   Default <none>
9.130.3.4       <direct>        ST01     1500    HOST
9.130.3.9       <direct>        VMED     4096    HOST
9.130.3.12      9.130.48.134    ISRING   Default HOST
9.130.3.13      9.130.48.134    ISRING   Default HOST
9.130.3.26      <direct>        VMPERF   4096    HOST
9.0.0.0         <direct>        ISRING   2000    0.255.255.0     0.130.48.0
9.130.57.89     9.130.48.253    ISRING   Default HOST
9.130.58.10     9.130.48.253    ISRING   Default HOST
9.0.0.0         9.130.48.5      ISRING   Default 0.255.255.0     0.130.154.0
9.0.0.0         9.130.48.5      ISRING   Default 0.255.255.0     0.130.155.0
9.0.0.0         <direct>        ETRING   1500    0.255.255.0     0.130.176.0
9.130.249.33    9.130.48.134    ISRING   Default HOST
9.130.249.34    9.130.48.134    ISRING   Default HOST
9.130.249.35    9.130.48.134    ISRING   Default HOST
9.130.249.36    9.130.48.134    ISRING   Default HOST
9.130.249.37    9.130.48.134    ISRING   Default HOST
9.130.249.39    9.130.48.134    ISRING   Default HOST
125.0.0.0       9.130.176.110   ETRING   Default <none>
```

# Session Establishment

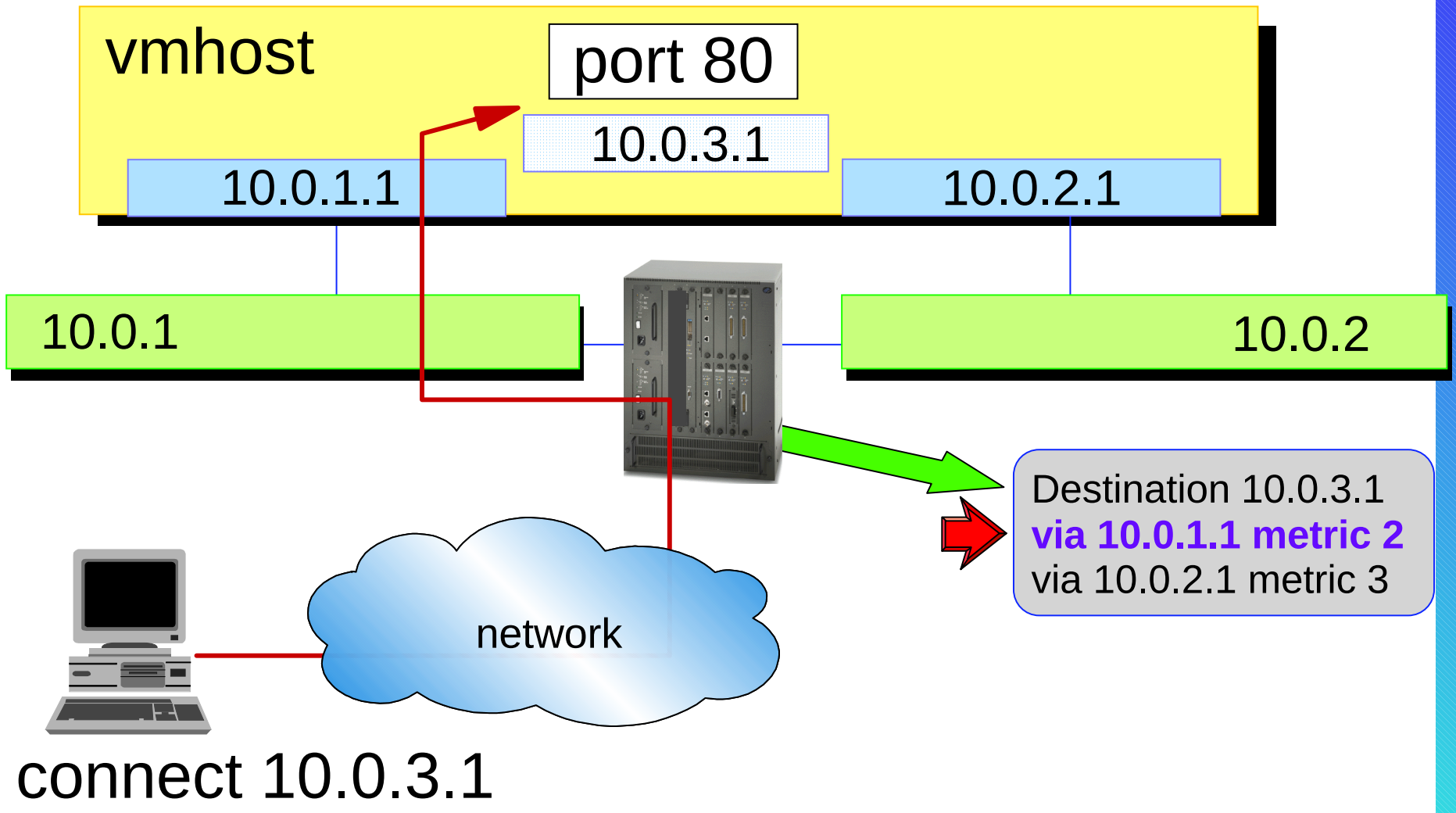


# Session Failure and Recovery

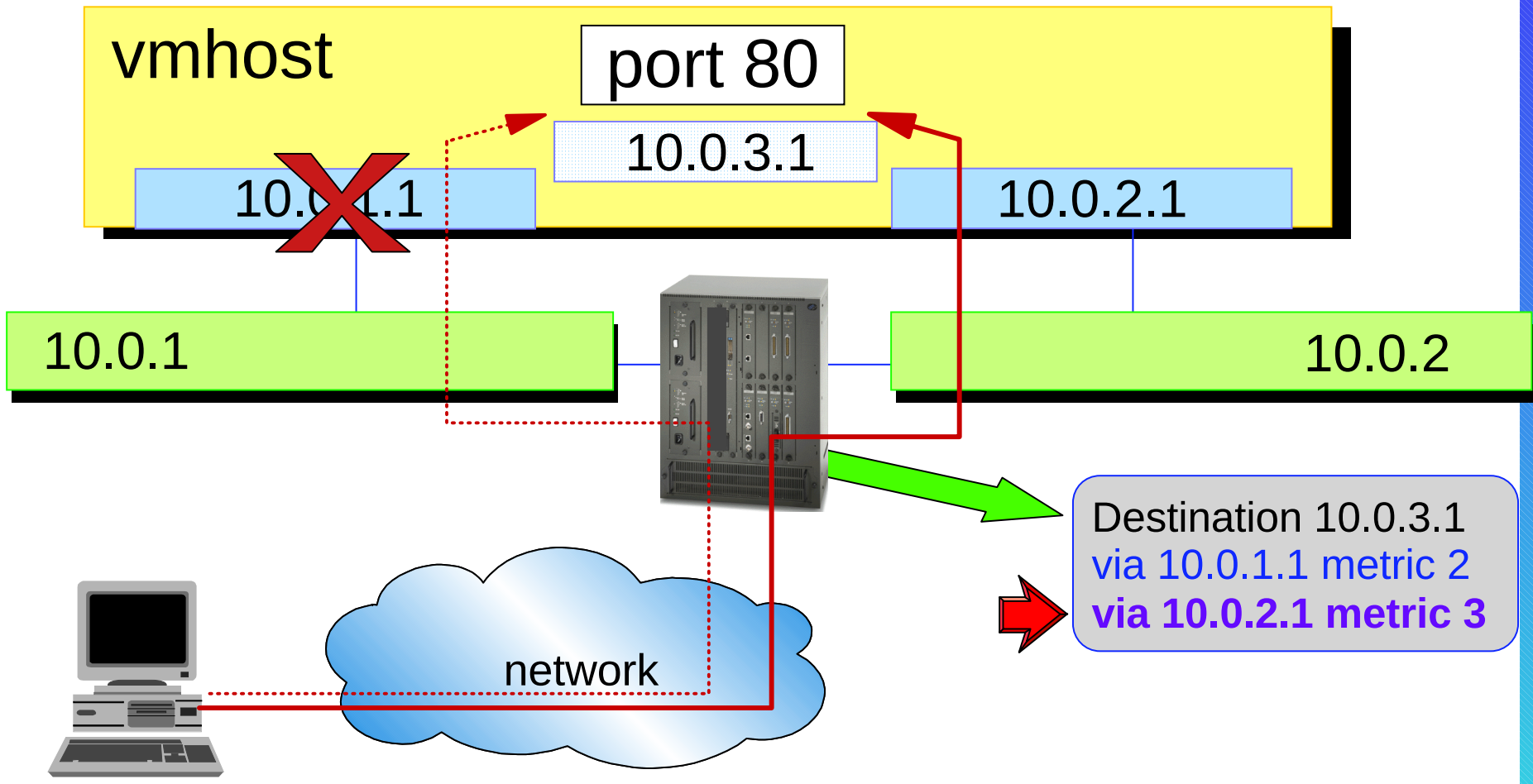




# Virtual IP Addressing



# Virtual IP Addressing



# VIPA: Virtual IP Addressing

- Insulates clients from IP address changes
- Protects clients from hardware outages
- Provides increased host availability
- Works best with RouteD

# Read More About It

- *VM TCP/IP Planning and Customization*, SC24-5847
- *TCP/IP Solutions for VM/ESA*, SG24-5459
  - http://www.redbooks.ibm.com
- *TCP/IP Illustrated, Vol. I*
  W. Richard Stevens, Addison Wesley
  ISBN 0-201-63346-9
- *Internetworking with TCP/IP*
  Douglas P. Comer, Prentice Hall
  ISBN 0-13-216987-8

# Contact Information


- By e-mail: Alan_Altmark@us.ibm.com
- In person: USA 607.752.6027
- On the Web: http://www.ibm.com/vm/devpages/altmarka
- Mailing lists: IBMTCP-L@vm.marist.edu
  VMESA-L@listserv.uark.edu
- On TalkLink: TCPIP CFORUM