Wireless LAN - CO72047

Prof W Buchanan - Centre for Dist. Computing and Security

Wireless LAN CO72047

Unit 7: Filtering

Prof. Bill Buchanan

Contact: w.buchanan@napier.ac.uk
Room: C.63
Telephone: X2759
MSN Messenger: w_j_buchanan@hotmail.com
WWW: http://www.dcs.napier.ac.uk/~bill
http://buchananweb.co.uk

INVESTOR IN PEOPLE

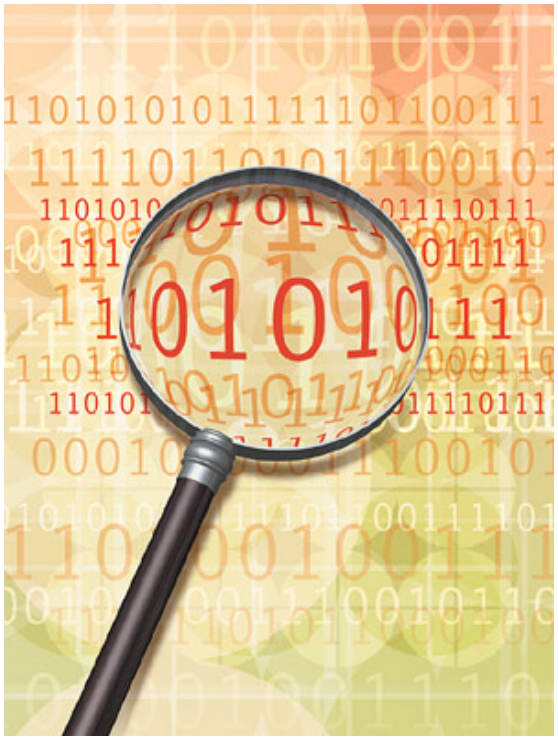
NAPIER UNIVERSITY
EDINBURGH

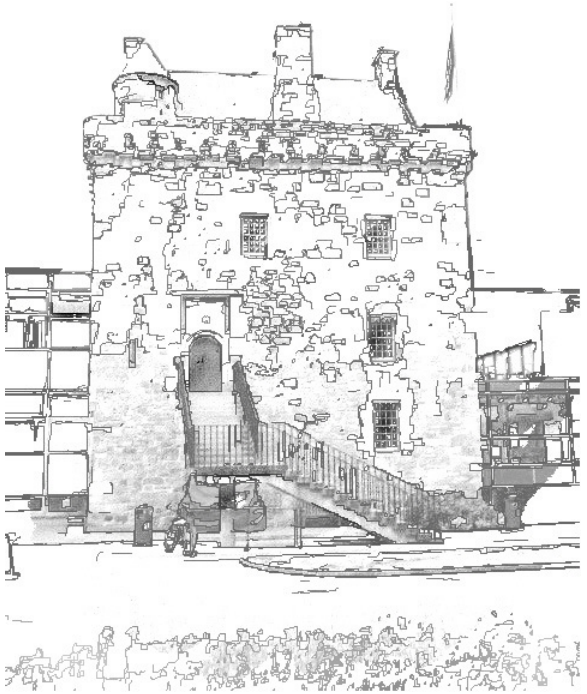
Author: Bill Buchanan

Wireless LAN - CO72047

Prof W Buchanan - Centre for Dist. Computing and Security

Module Descriptor





INVESTOR IN PEOPLE

NAPIER UNIVERSITY
EDINBURGH

Author: Bill Buchanan

Wireless LAN - CO72047

Prof W Buchanan - Centre for Dist. Computing and Security

Week	Date	Academic	Cisco	Lab/Tutorial
1	1 Oct	1**: Radio Wave Fundamentals**		
2	8 Oct	**2: Wireless Fundamentals**	Intro to Wireless LANs	Lab 1/2: Access Point Tutorial 1 (T)
3	15 Oct	**3: Ad-hoc and Infrastructure Networks**	IEEE 802.11 and NICs	Lab 3: Ad-hoc Networks
4	22 Oct	**4: Encryption**	Wireless Radio Technology	Lab 4: Infrastructure Networks
5	29 Oct	**5: Authentication**	Wireless Topologies	Lab 5: Remote Connections
6	5 Nov	**6: Antennas**	Access Points	Lab 6: Encryption/Authen
7	12 Nov	7: Filtering/8. VLANs	Bridges	Lab 7: Filter
8	19 Nov	**Napier Test (40%)**	Antennas	Lab 8: VLAN
9	26 Nov		Security	Lab 9: VLAN/802.1Q
10	3 Dec	Cisco Academy/Additional Material	Applications	Lab 10: IP Routing
11	10 Dec	Cisco Academy /Additional Material	Site Survey	Lab 11: RADIUS
12	17 Dec	Cisco Academy /Additional Material	Troubleshooting	Lab 12: SNMP
Holidays				
13	7 Jan	Revision/Cram (Cisco Exam)	Emerging Technologies	**Coursework/Practical (50%)**
14	14 Jan	Revision/Cram (Cisco Exam)	**Cisco Exam (10%)**	
15	21 Jan			

Author: Bill Buchanan

SWSCUST Module Individual - Windows Internet Explorer

http://timetableing.napier.ac.uk/reporting/individual;module;id;co72047%0D%0A?days=1-7&weeks=1-13;16-18&peric

File Edit View Favorites Tools Help

SWSCUST Module Individual

Module: co72047 - Wireless LANs

Weeks: 0-12, 13-15 (24 Sep 2007-27 Jan 2008)

	9:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00
Mon												
Tue												
Wed	Prof William Buchanan Merch.F27 co72047.L01 1-12, 13		Prof William Buchanan, Mr Jamie Graves, Mr Merch.co.C6 Lionel Saliou co72047.P03 1-12, 13-15	Prof William Buchanan, Mr Jamie Graves, Mr Merch.co.C6 Lionel Saliou co72047.P01 1-12, 13-15		Prof William Buchanan, Mr Jamie Graves, Mr Merch.co.C6 Lionel Saliou co72047.P02 1-12, 13-15			Prof William Buchanan, Mr Jamie Graves, Mr Merch.co.C6 Lionel Saliou co72047.P04 1-12, 13-15			
Thu	Merch.co.C6 co72047.U01 1-12, 13-15											
Fri												
Sat												
Sun												

<< Back Print Timetable Date/Time: 1 Oct 2007 21:49

Template: SWSCUST Module Individual

Done Internet 100%

Wireless LAN co72047
Prof W Buchanan
for Dist. Computi

Author: Bill Buchanan

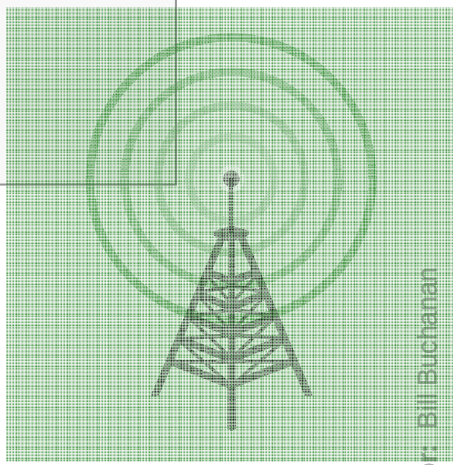
Wireless connections … which technology?

Areas covered:

Filtering.

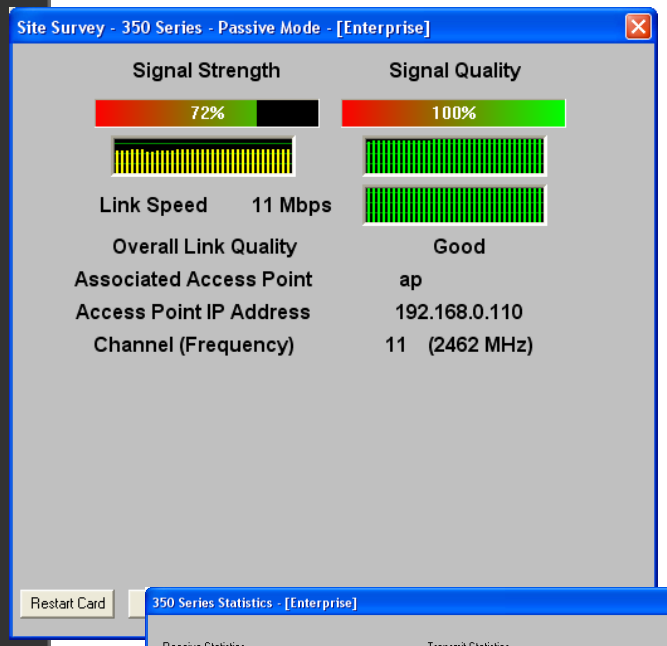
ACLs.

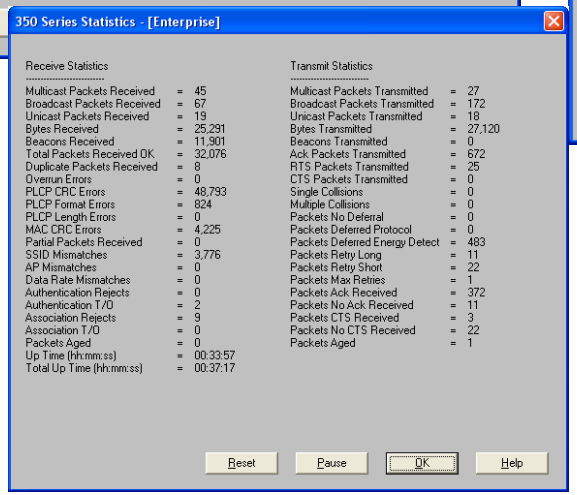
MAC address filtering.

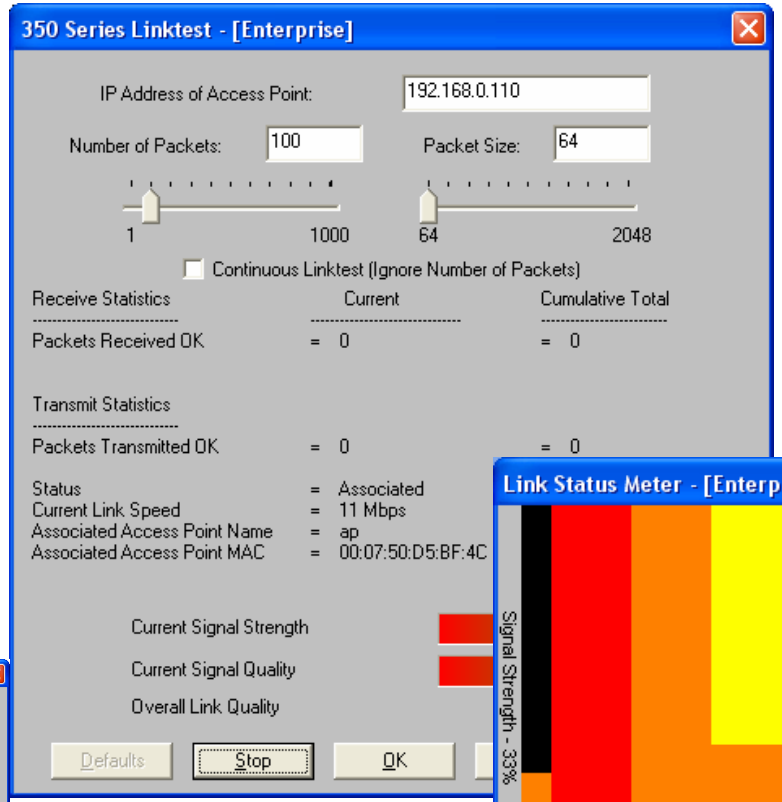


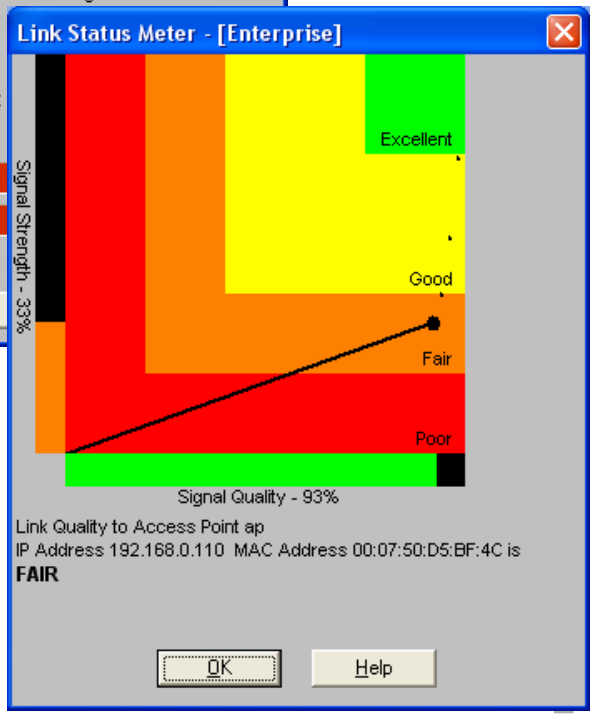
Wireless LAN - CO72047

Prof W Buchanan - Centre for Dist. Computing and Security



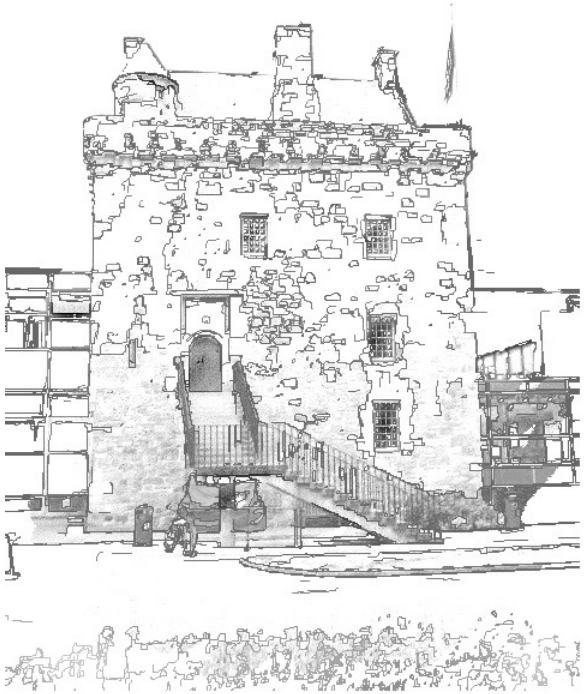






Author: Bill

Background



Data encapsulation

E-mail application program

Application

Application program makes contact with network application for e-mail

```
Hello.
Fred.
```

Presentation

Convert data into a form which can be transmitted

```
To: Fred
From: Bert
Hello.
Fred.
```

Session

Contact remote system and request a transmission

```
HELO sys.com
FOR Fred
To: Fred
From: Bert
Hello.
Fred.
```

Transport

Negotiate data transfer and split data into segments

Start	Addr	Seg 1	End
Start	Addr	Seg 2	End
Start	Addr	Seg 3	End

Seg 1
Seg 2
Seg 3

Network

Add source and destination addresses

Addr	Seg 1
Addr	Seg 2
Addr	Seg 3

Data link

Data packet converting into a form which can be transmitted over the network

Physical

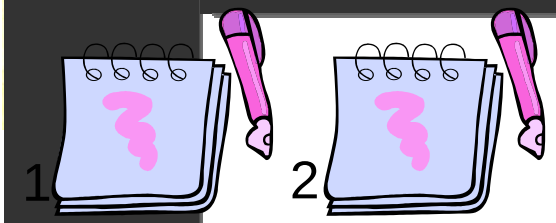
The data frame is converted into binary form and transmitted over a physical connection

Wireless LAN - CO72047

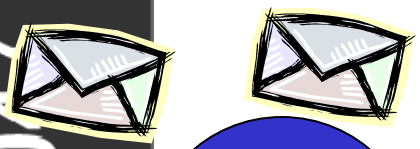
Prof W Buchanan - Centre for Dist. Computing and Security

Author: Bill Buchanan

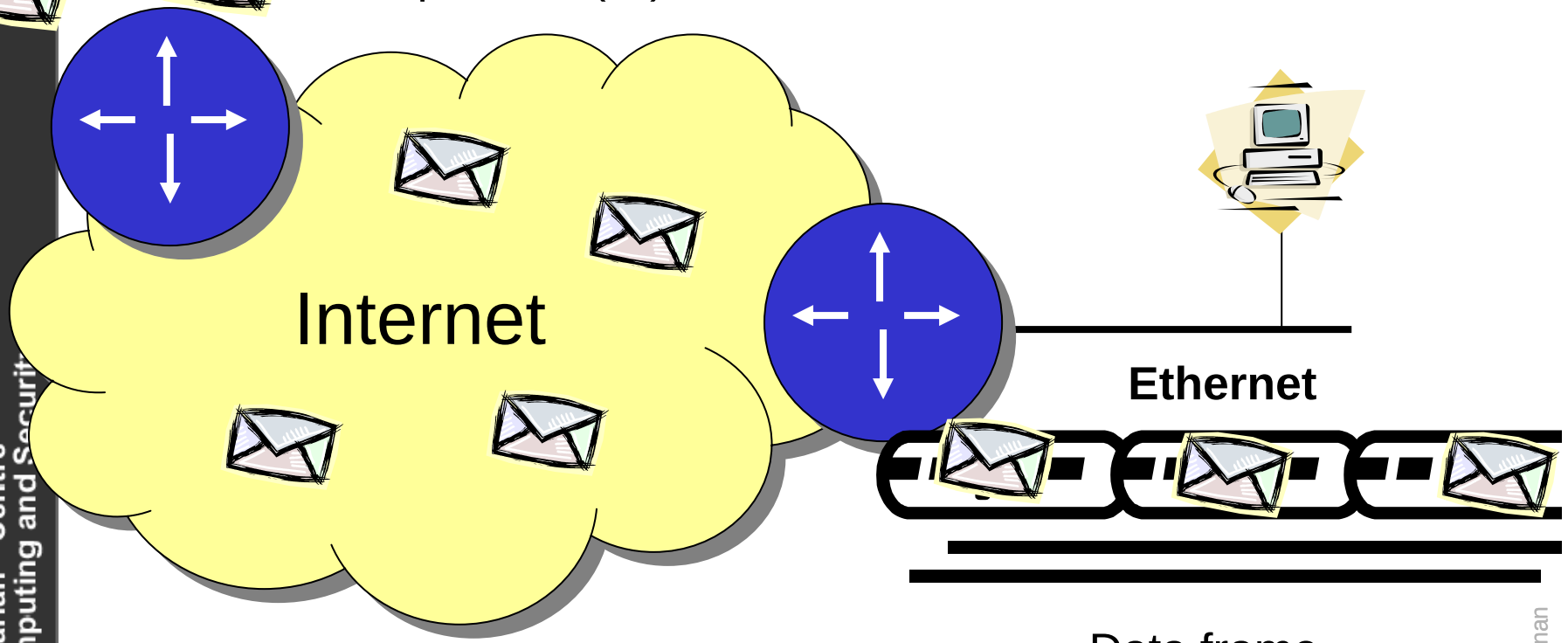
Terms for each layer



Data segments (TCP)



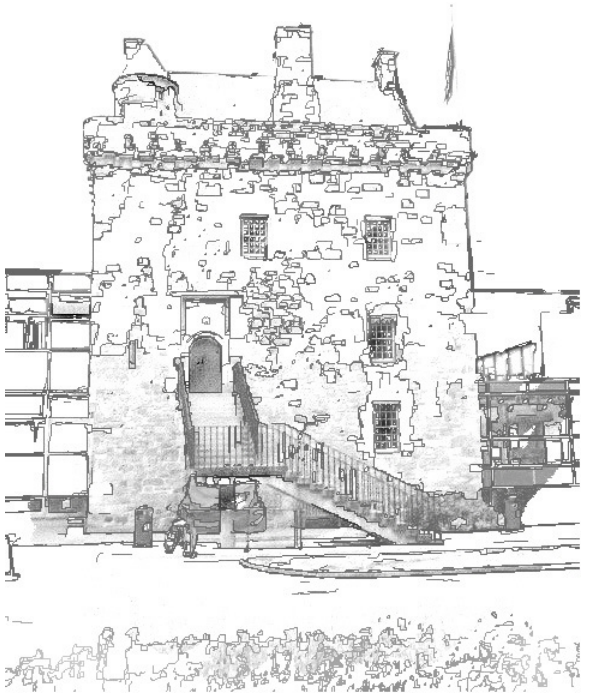
Data packets (IP)



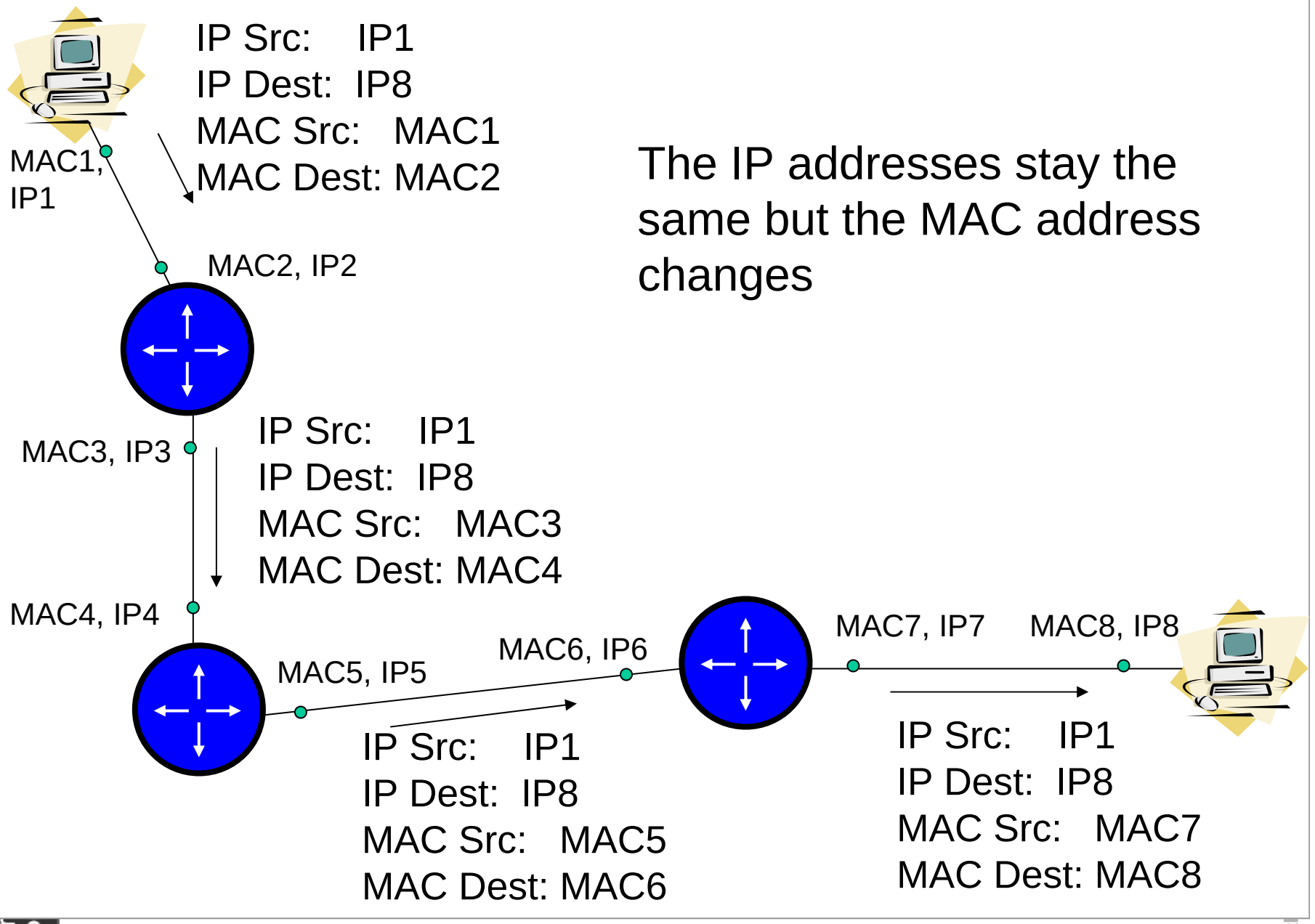
Data frame

Author: Bill Buchanan

IP and TCP ... The Greatest of the Protocols!

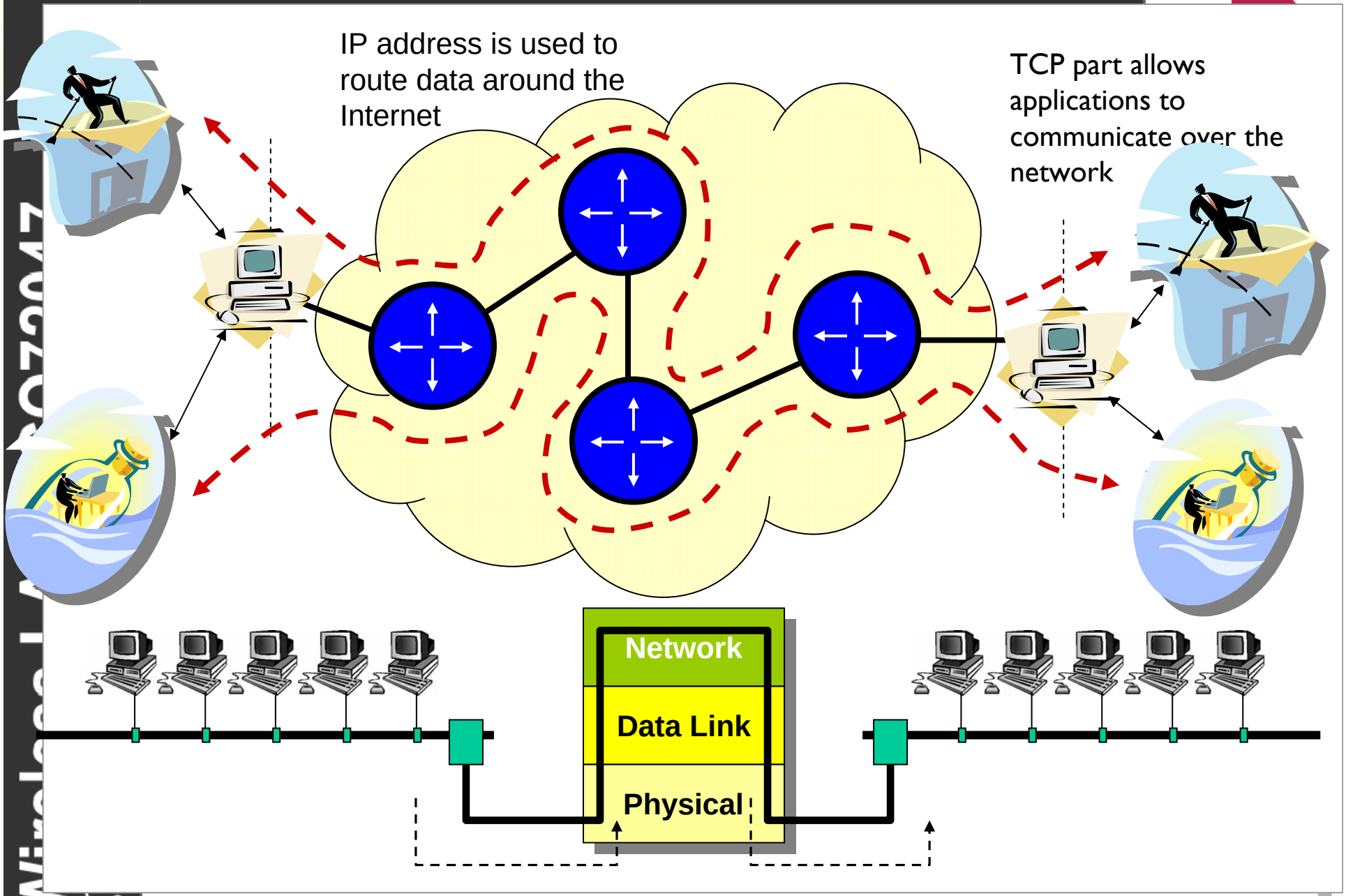


IP and MAC Addresses

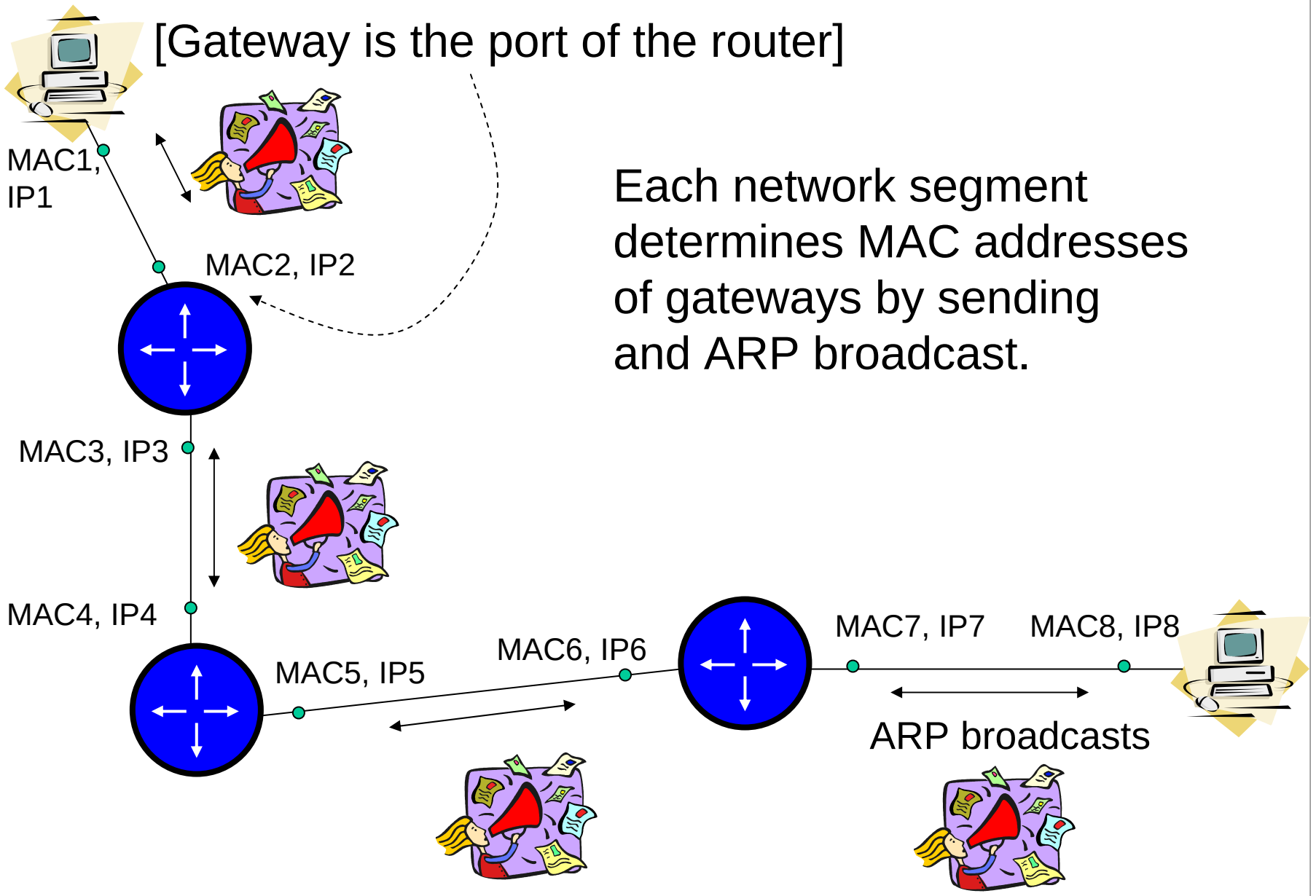


The IP addresses stay the same but the MAC address changes

IP and TCP

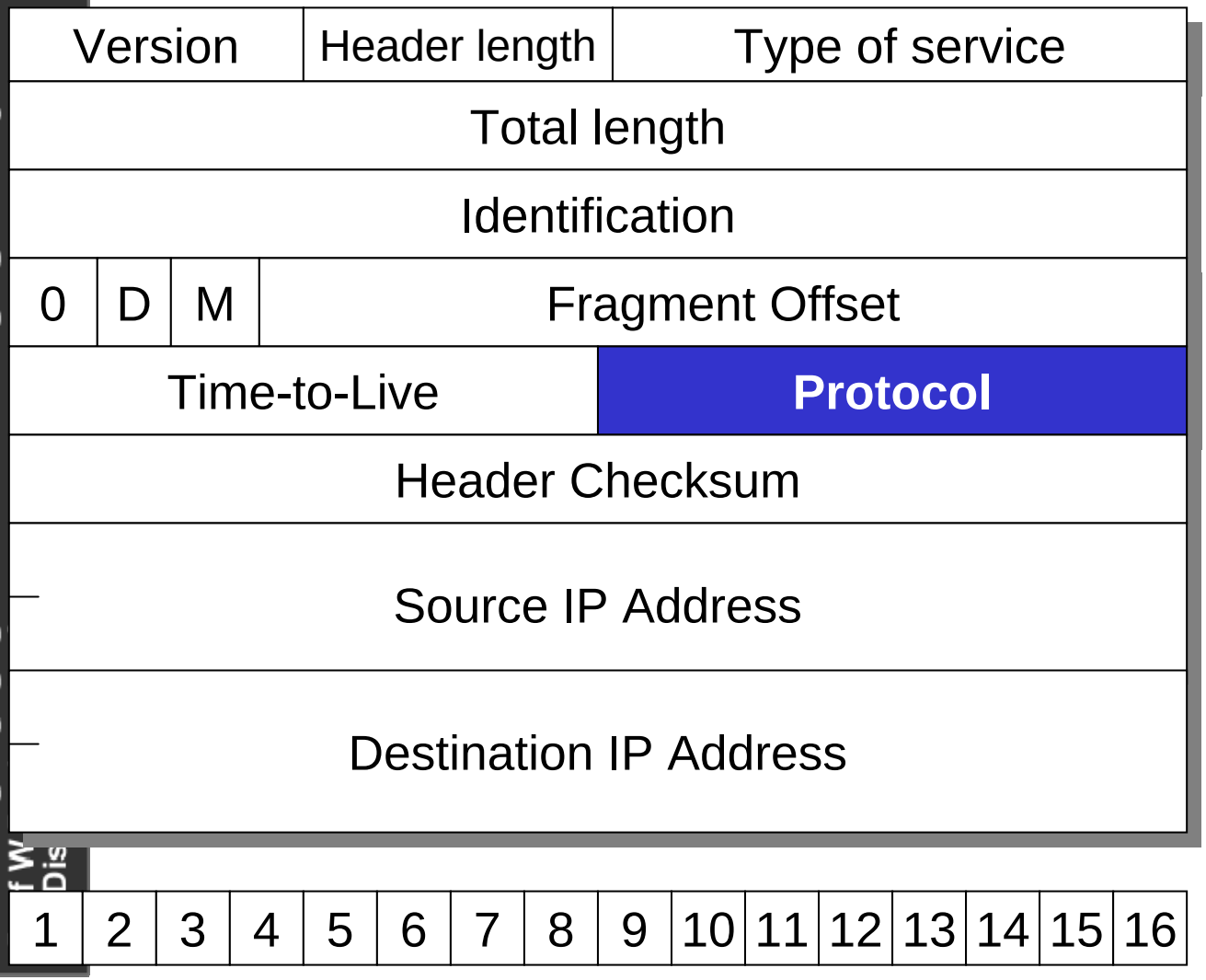

IP address is used to route data around the Internet
TCP part allows applications to communicate over the network
Network
Data Link
Physical

IP and MAC Addresses



IP header

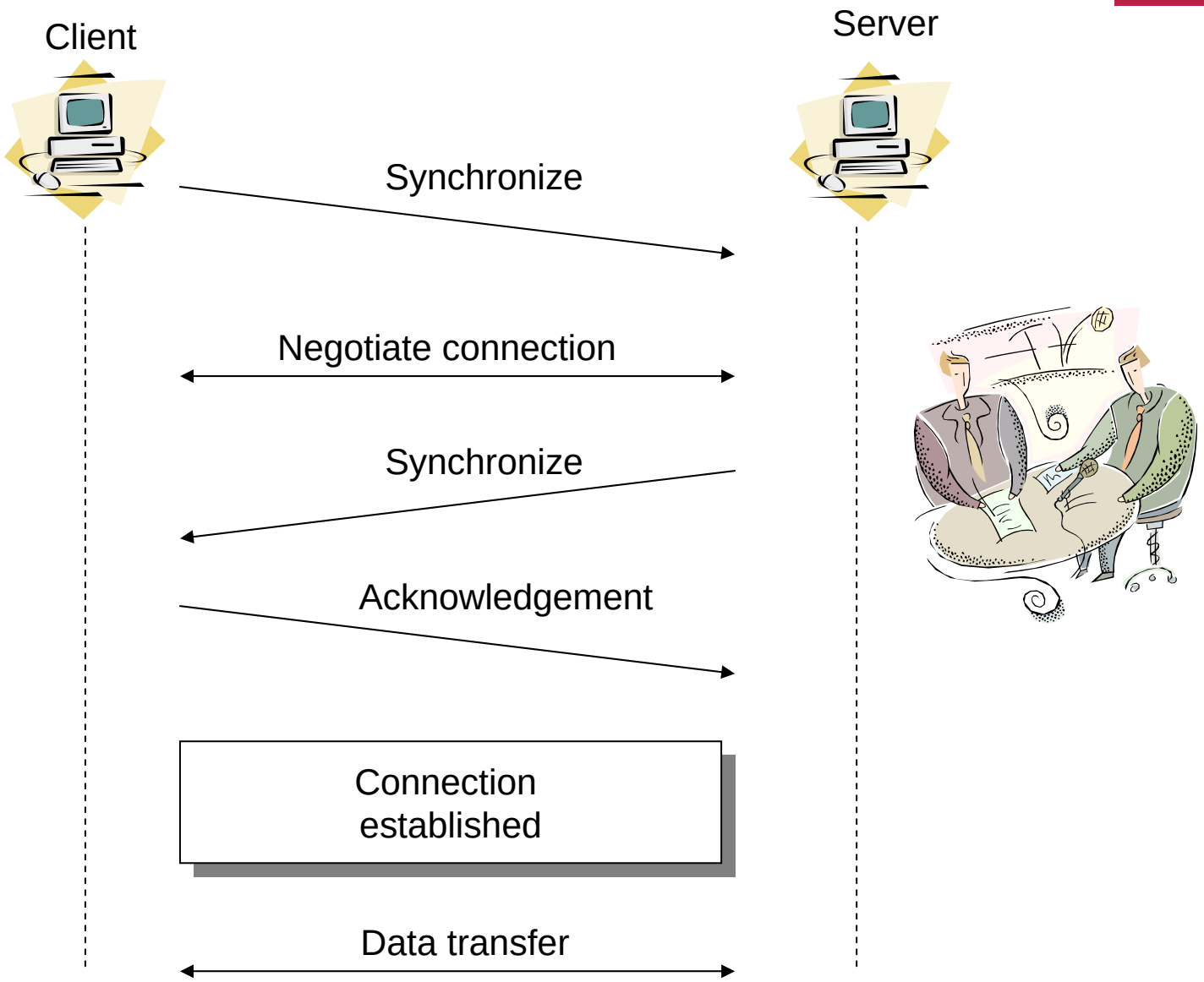
Data Packet



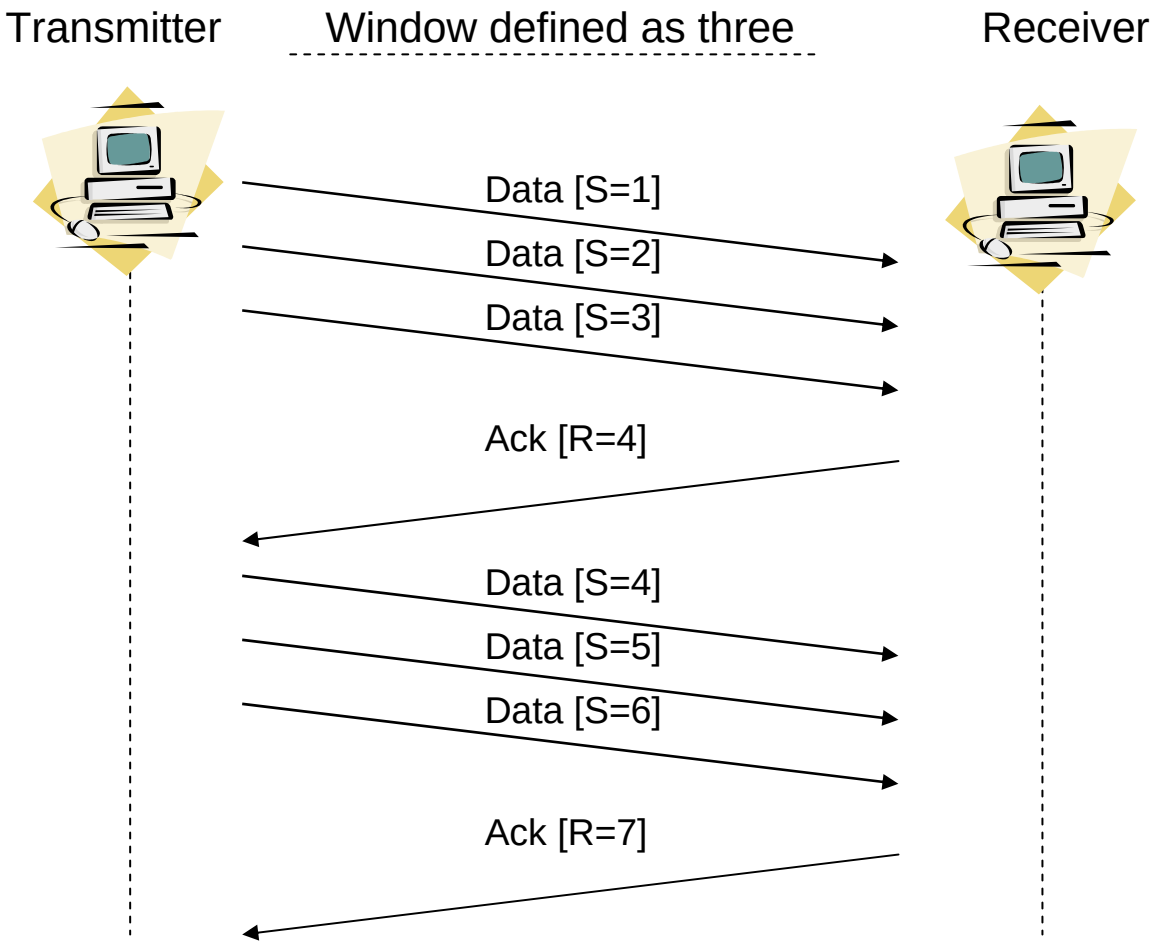
Protocol (8 bits). Different transport protocols can be used on the datagram. The 8-bit protocol field defines the type to be used. E.g. 1 – ICMP and 6 – TCP.

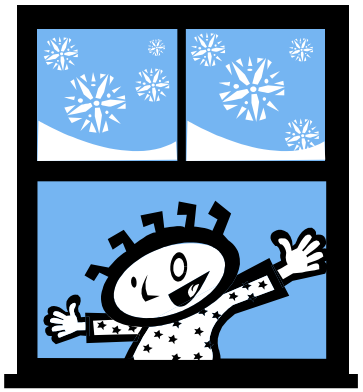
Author: Bill Buchanan

Synchronization and acknowledgement



Windowing

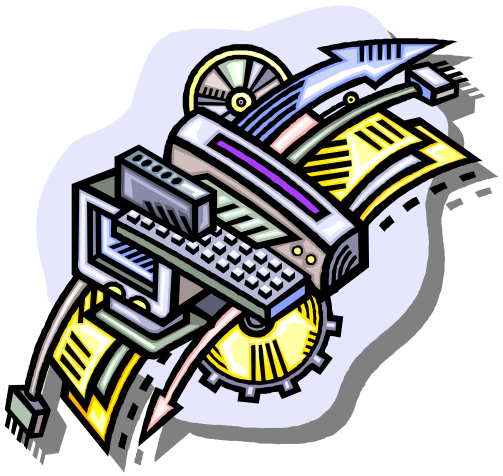

Transmitter
Window defined as three
Receiver
Data [S=1]
Data [S=2]
Data [S=3]
Ack [R=4]
Data [S=4]
Data [S=5]
Data [S=6]
Ack [R=7]



TCP operation

Client uses a unique local software port for the connection

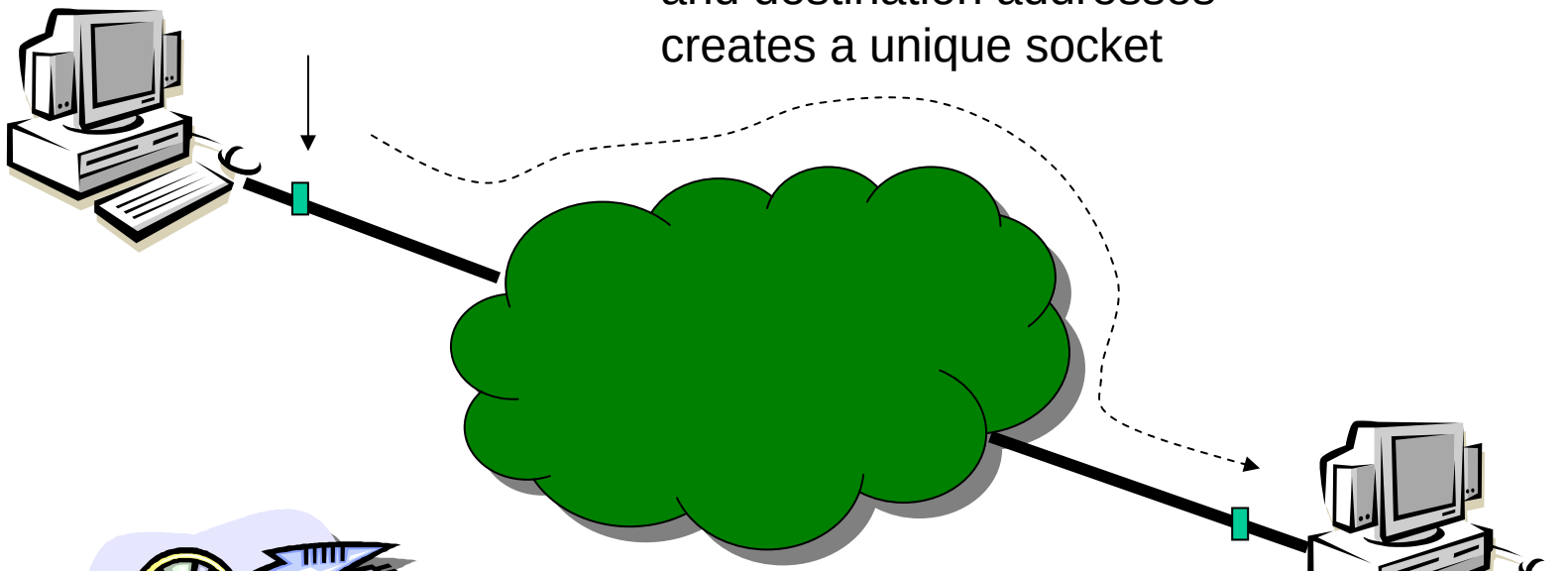
Client makes a connection with its own local port and the destination port of the server

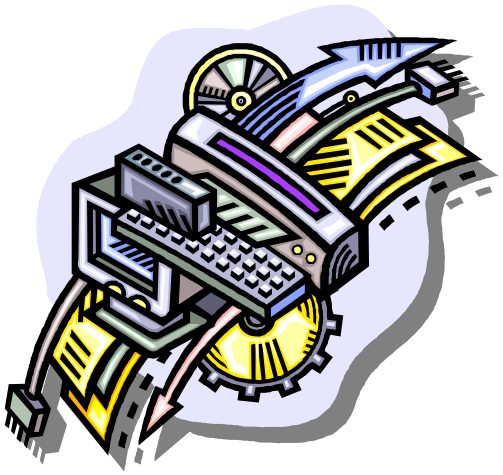


Server listens on a given port. Every client who wants this service connects to this port

TCP operation

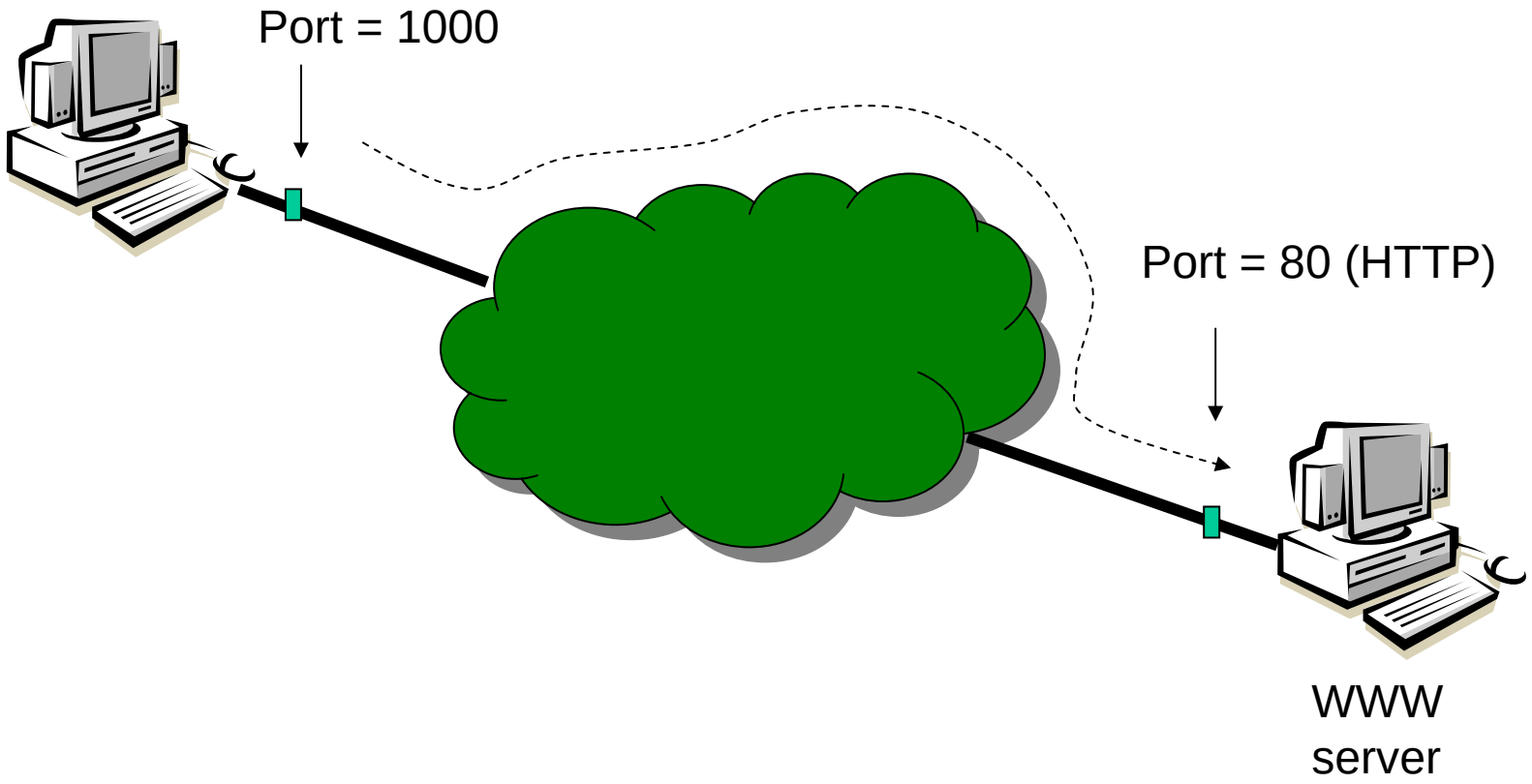
The binding of the local and destination port, and the local and destination addresses creates a unique socket



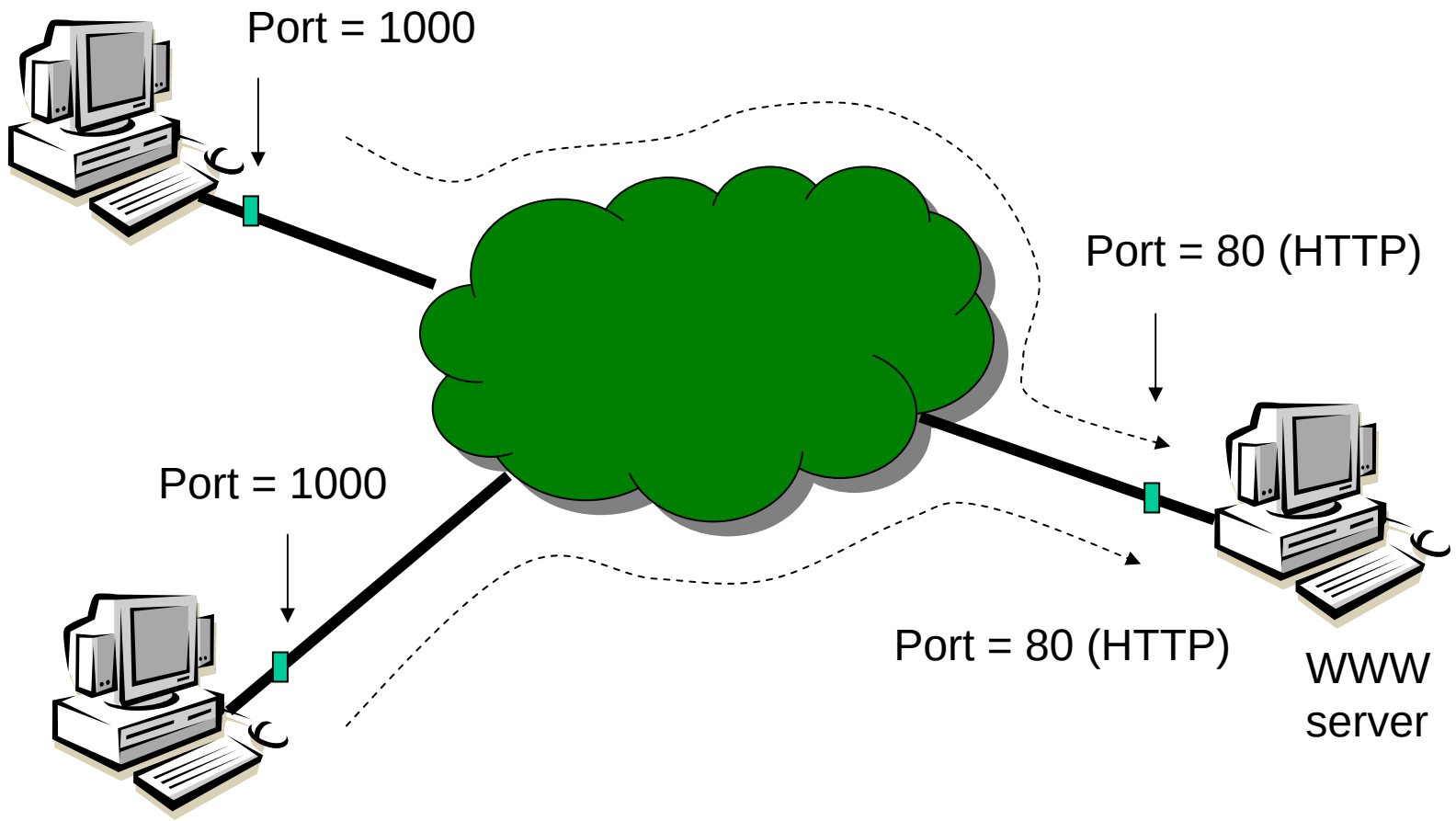


Sockets allow application programs to send data to a socket interface

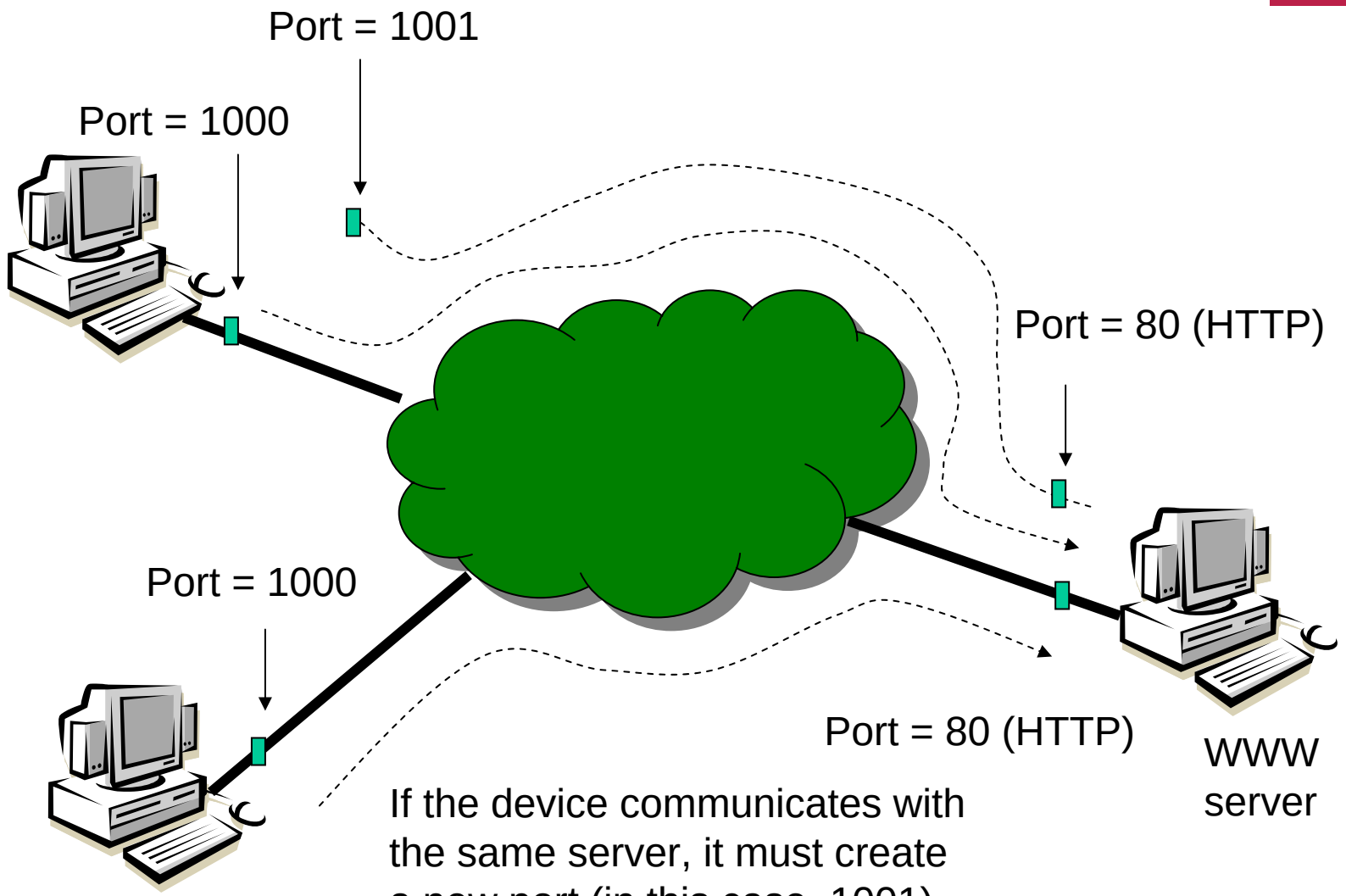
Example



Example

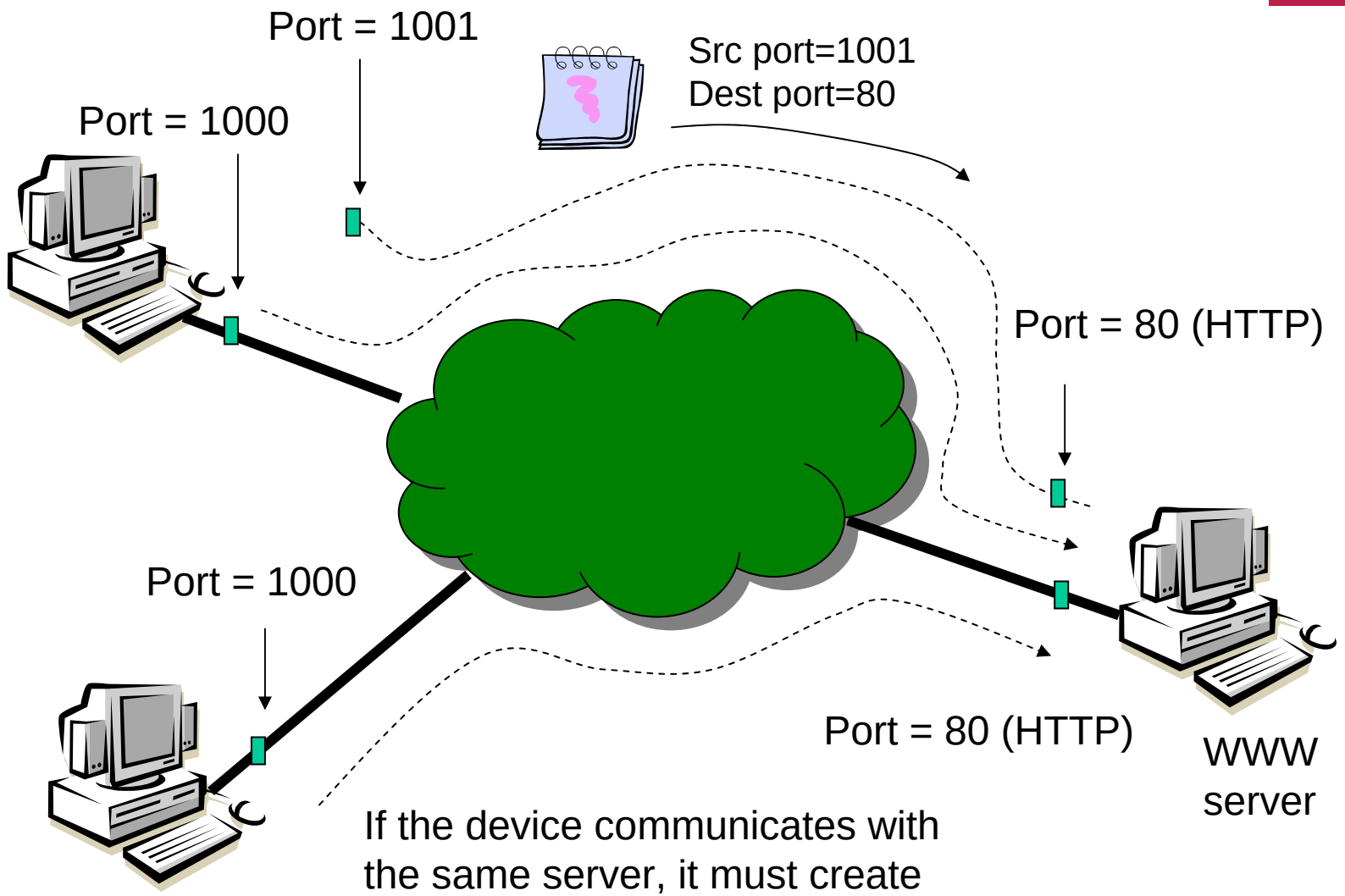


Example



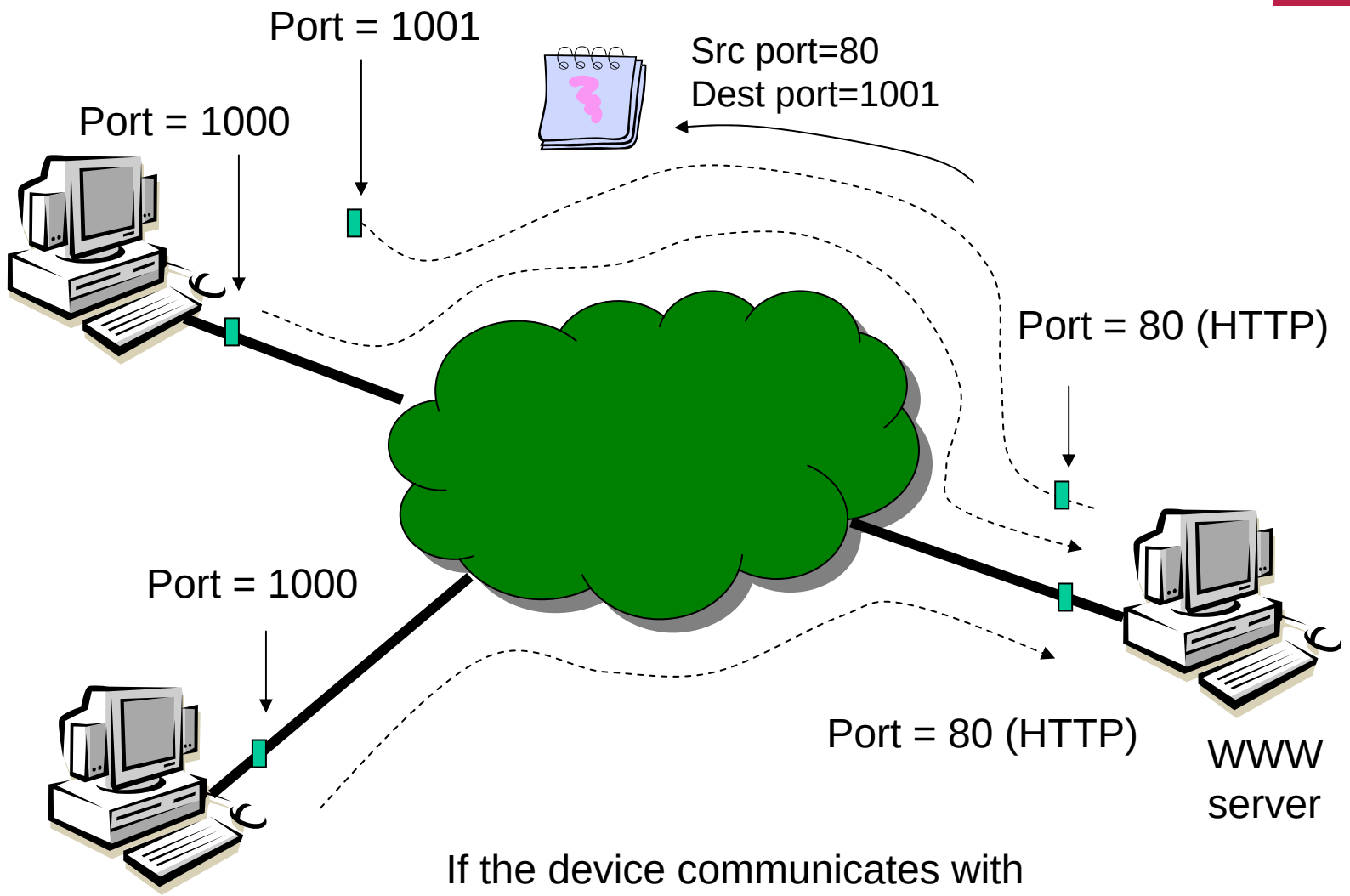
If the device communicates with the same server, it must create a new port (in this case, 1001)

Example



If the device communicates with the same server, it must create a new port (in this case, 1001)

Example



If the device communicates with the same server, it must create a new port (in this case, 1001)

TCP header

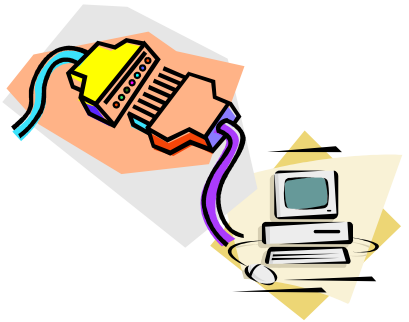
IP	TCP	Higher-level protocol/data

Source port	
Destination port	
Sequence number	
Acknowledgement number	
Data offset	Reserved/Flags
Window	
Checksum	
UrgentPtr	

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

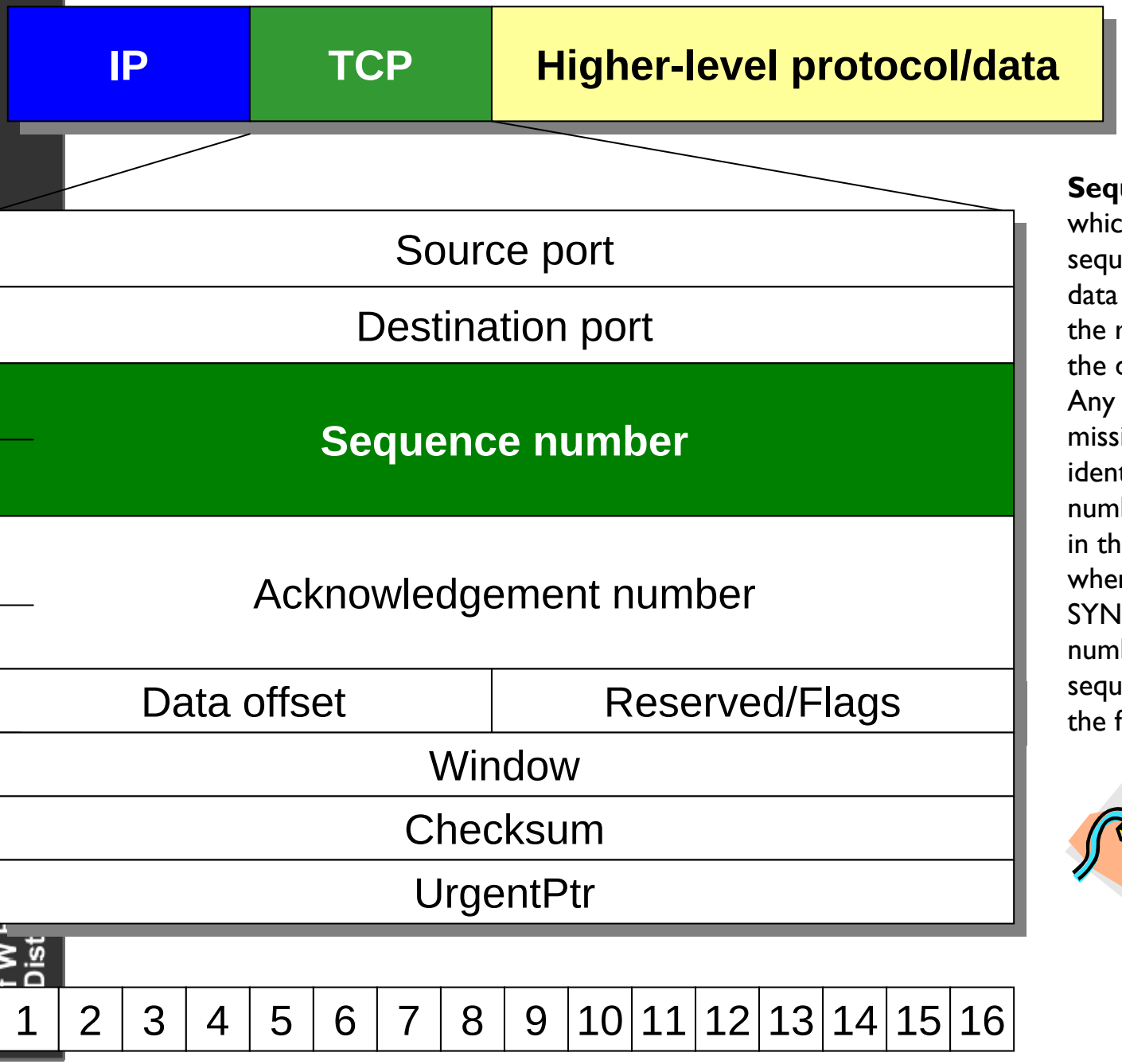
Data Packet

Source and destination port number – which are 16-bit values that identify the local port number (source number and destination port number or destination port).



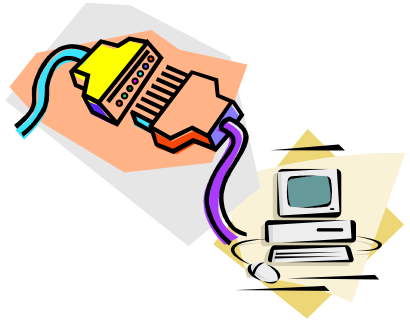
Author: Bill Buchanan

TCP header



Data Packet

Sequence number – which identifies the current sequence number of the data segment. This allows the receiver to keep track of the data segments received. Any segments that are missing can be easily identified. The sequence number of the first data byte in this segment (except when SYN is present). If SYN is present the sequence number is the initial sequence number (ISN) and the first data octet is ISN+1.



Author: Bill Buchanan

TCP header

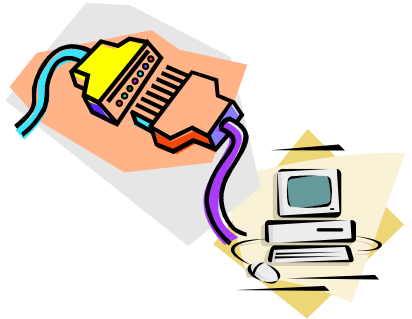
IP	TCP	Higher-level protocol/data

Data Packet

Source port	
Destination port	
Sequence number	
Acknowledgement number	
Data offset	Reserved/Flags
Window	
Checksum	
UrgentPtr	

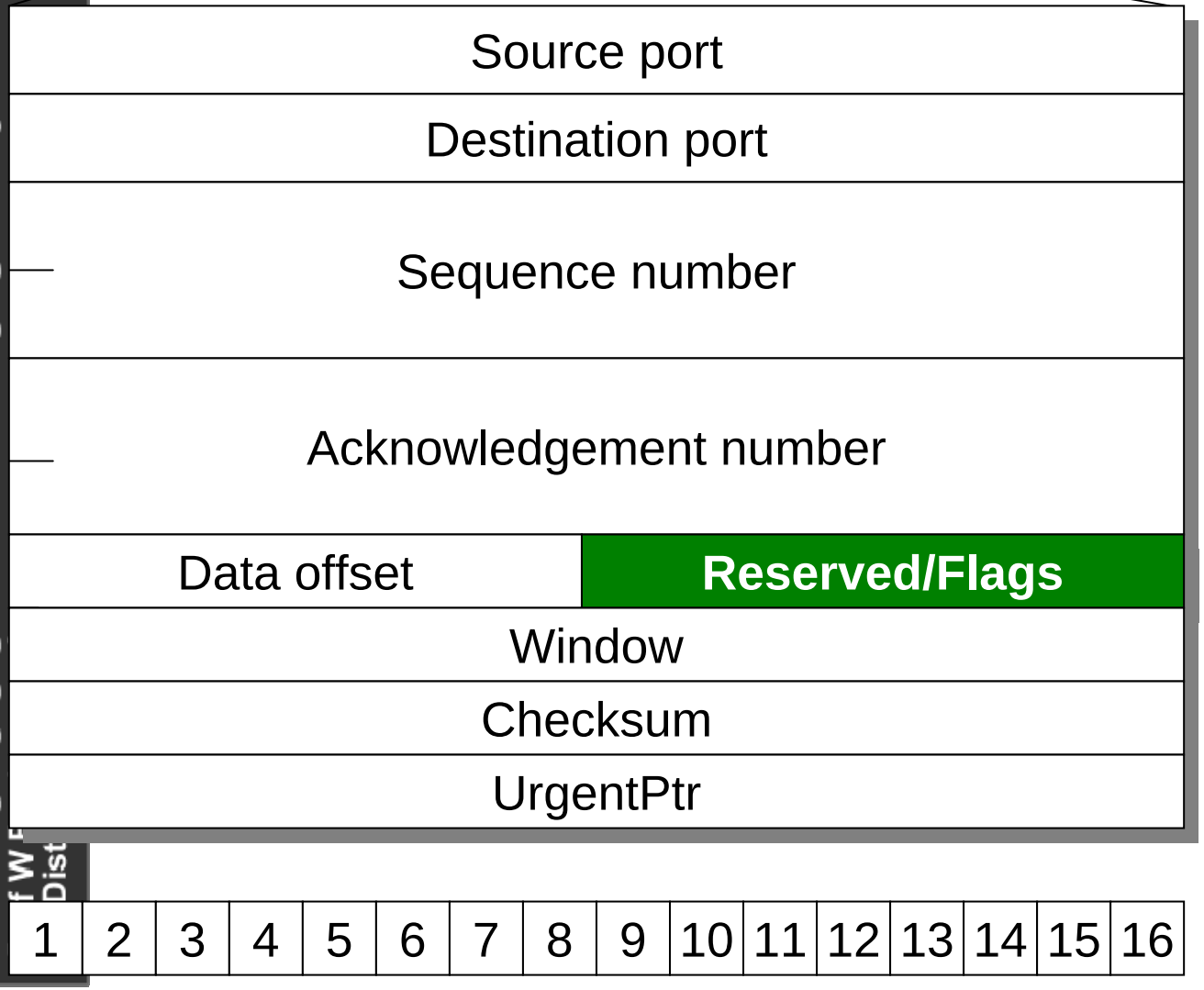
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Acknowledgement number – when the ACK bit is set, it contains the value of the next sequence number the sender of the packet is expecting to receive. This is always set after the connection is made.



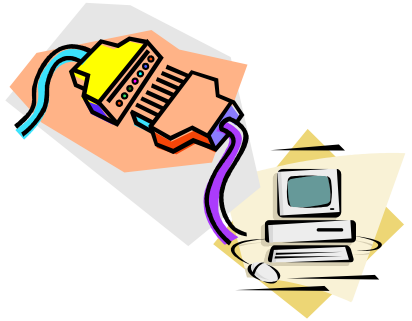
Author: Bill Buchanan

TCP header



Data Packet

Flags – the flag field is defined as UAPRSF, where U is the urgent flag (URG), A the acknowledgement flag (ACK), P the push function (PSH), R the reset flag (RST), S the sequence synchronise flag (SYN) and F the end-of-transmission flag (FIN).



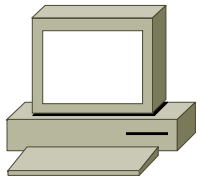
Author: Bill Buchanan

Examples of Data Exchange

Originator				Recipient
1.	CLOSED			LISTEN
2.	SYN-SENT	→	<SEQ=999><CTL=SYN>	SYN-RECEIVED
3.	ESTABLISHED	<SEQ=100><ACK=1000><CTL=SYN,ACK>	←	SYN-RECEIVED
4.	ESTABLISHED	→	<SEQ=1000><ACK=101><CTL=ACK>	ESTABLISHED
5.	ESTABLISHED	→	<SEQ=1000><ACK=101><CTL=ACK><DATA>	ESTABLISHED

	Originator			Recipient
1.	CLOSED			LISTEN
2.	SYN-SENT	→	<SEQ=999><CTL=SYN>	
3.	(duplicate)	→	<SEQ=900><CTL=SYN>	
4.	SYN-SENT	<SEQ=100><ACK=901>	<CTL=SYN,ACK>←	SYN-RECEIVED
5.	SYN-SENT	→	<SEQ=901><CTL=RST>	LISTEN
	(packet 2 received)	→		
7.	SYN-SENT		<SEQ=100><ACK=1000><CTL=SYN,ACK>←	SYN-RECEIVED
8.	ESTABLISHED	→	<SEQ=1000><ACK=101><CTL=ACK><DATA>	ESTABLISHED

Author: Bill Buchanan



SYN

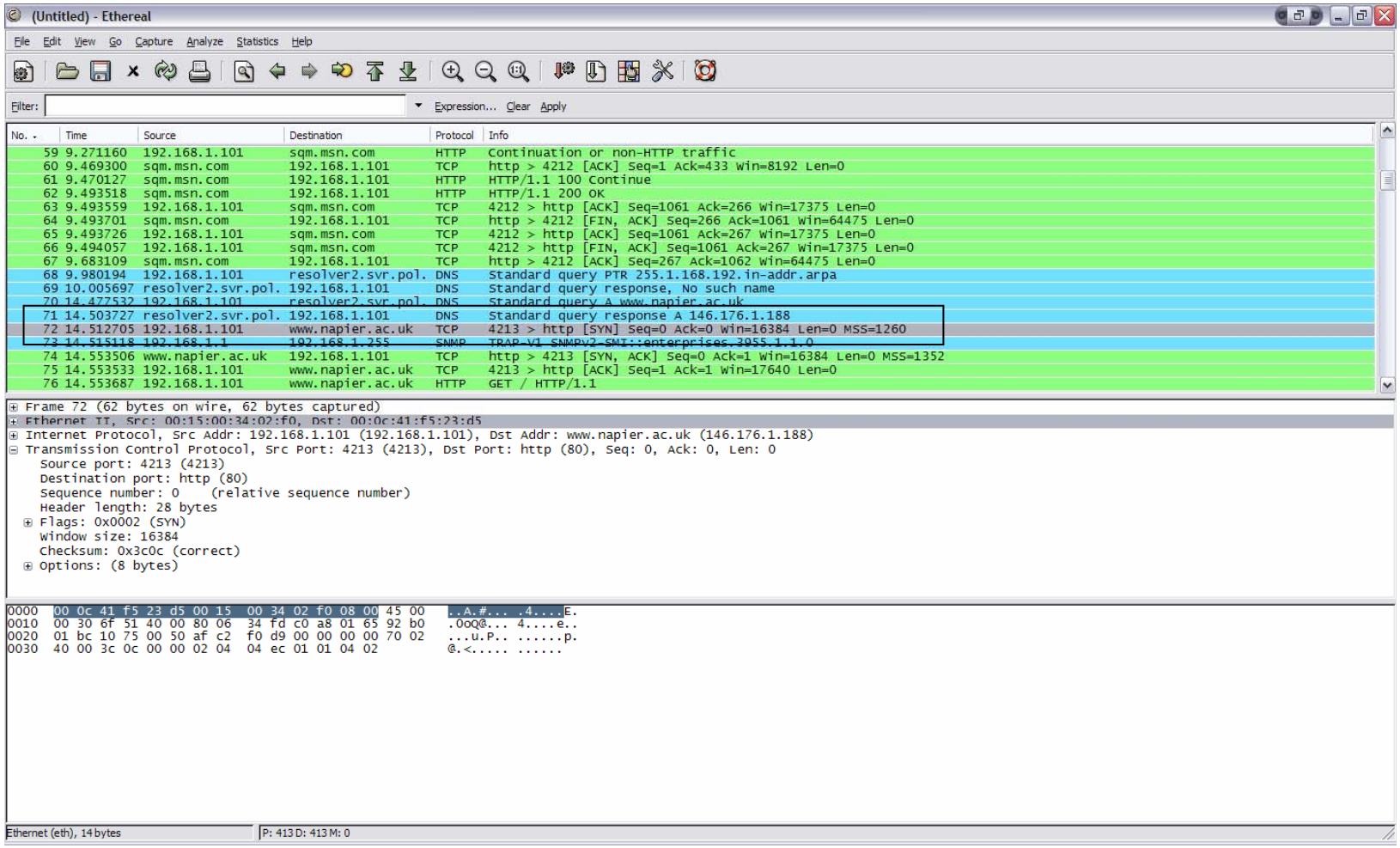
SYN, ACK

ACK

Client

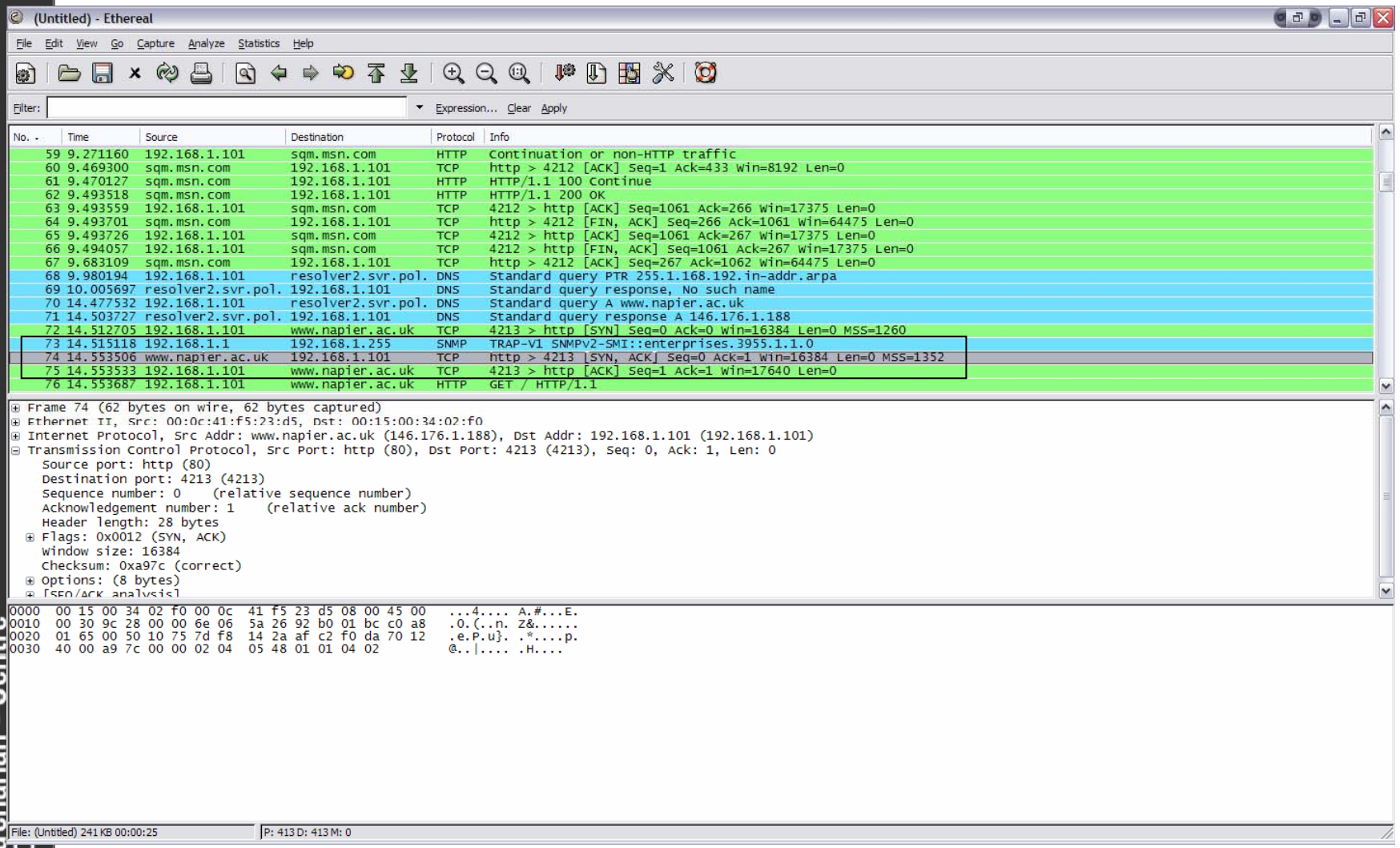
Server

SYN



Author: Bill Buchanan

SYN, ACK

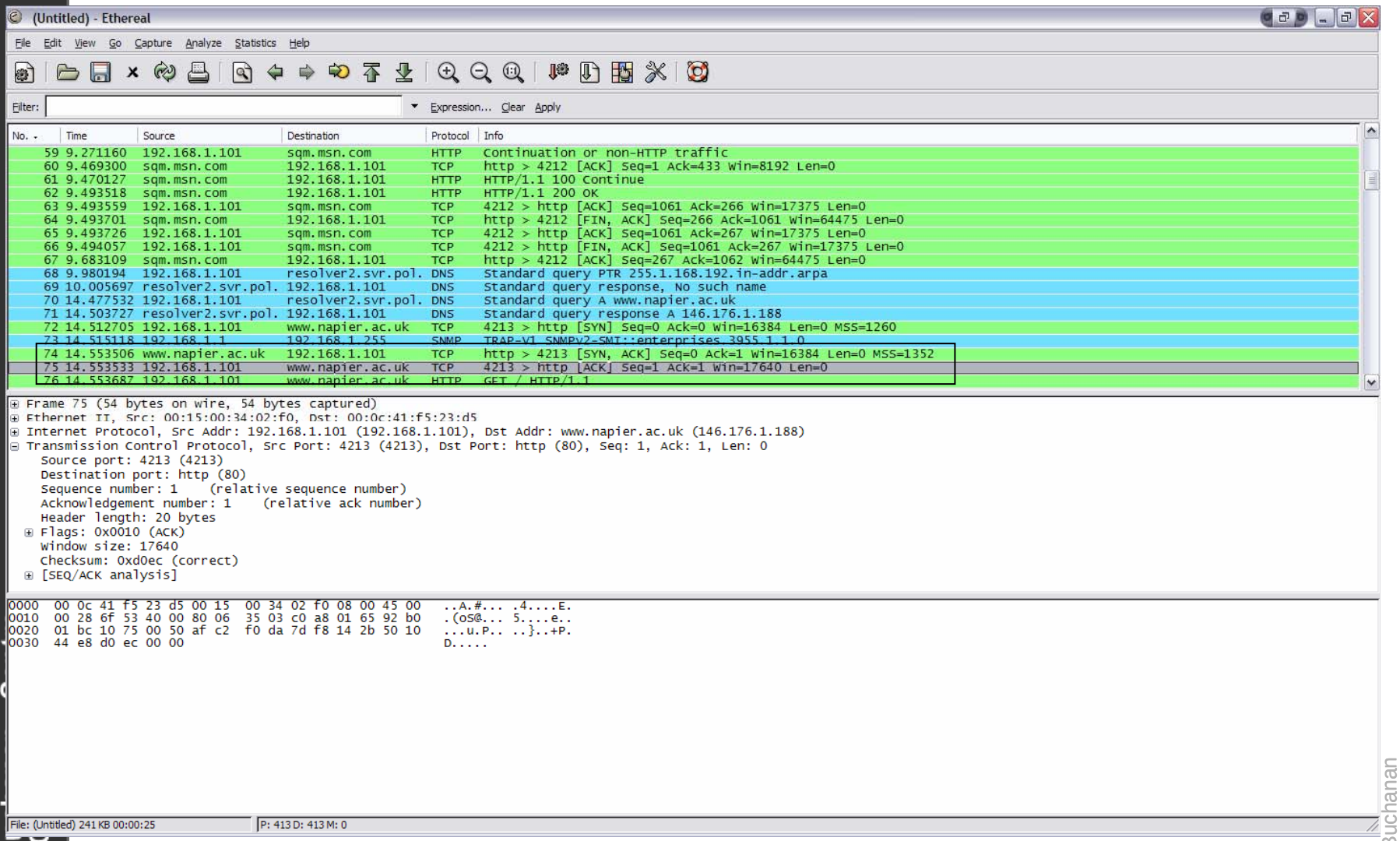


Wireless LAN - CO72047

Prof W Buchanan - Centre for Dist.

Author: Bill Buchanan

ACK



Wireless LAN - CO72047

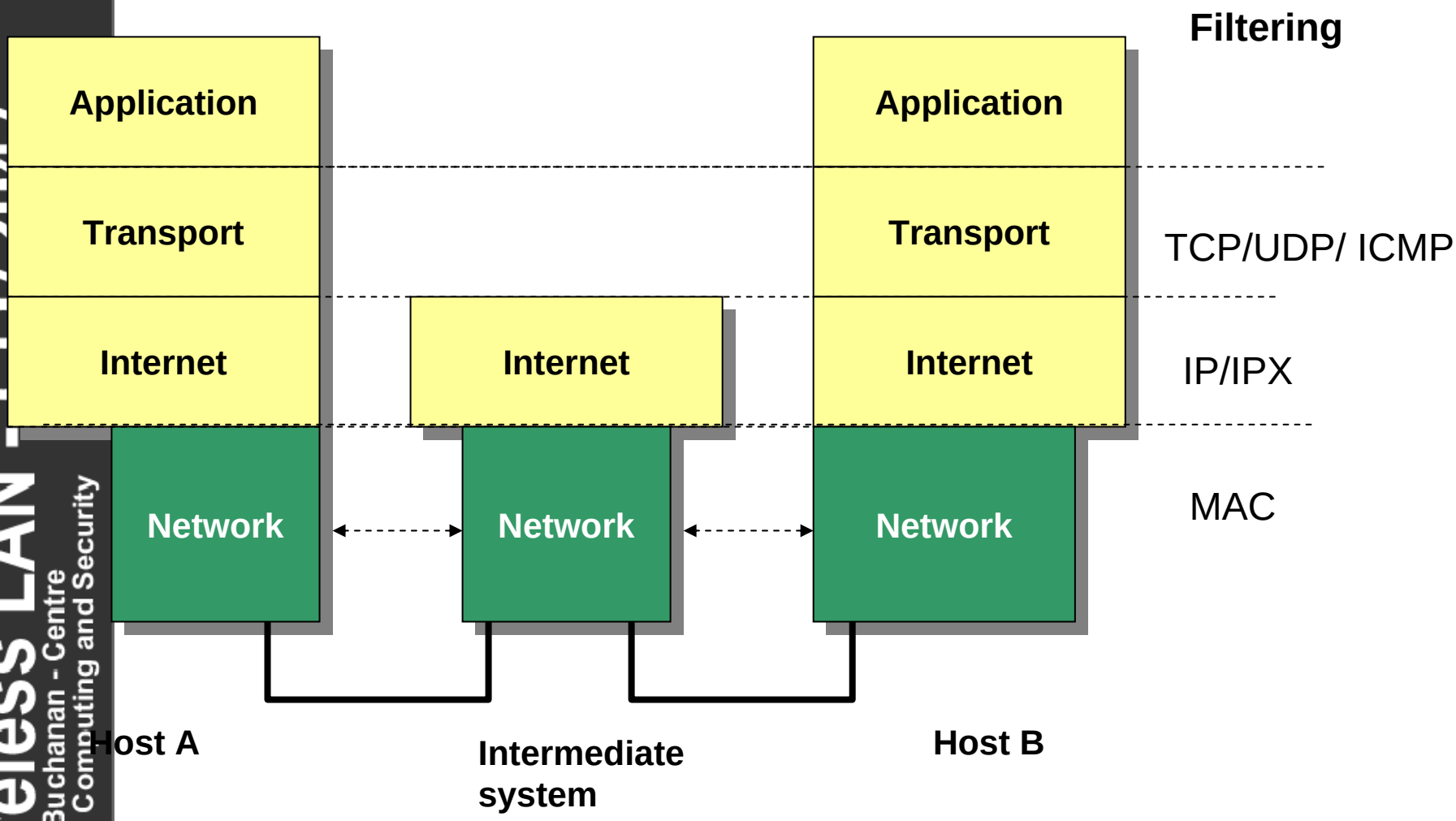
Prof W B
for Dist.

Author: Bill Buchanan

Filtering

Author: Bill Buchanan

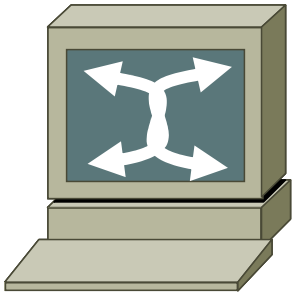
Layer filtering

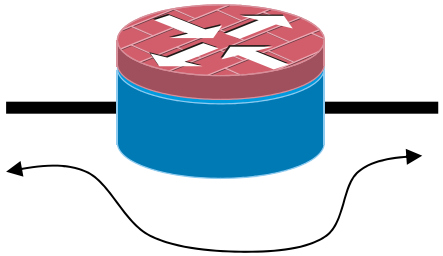


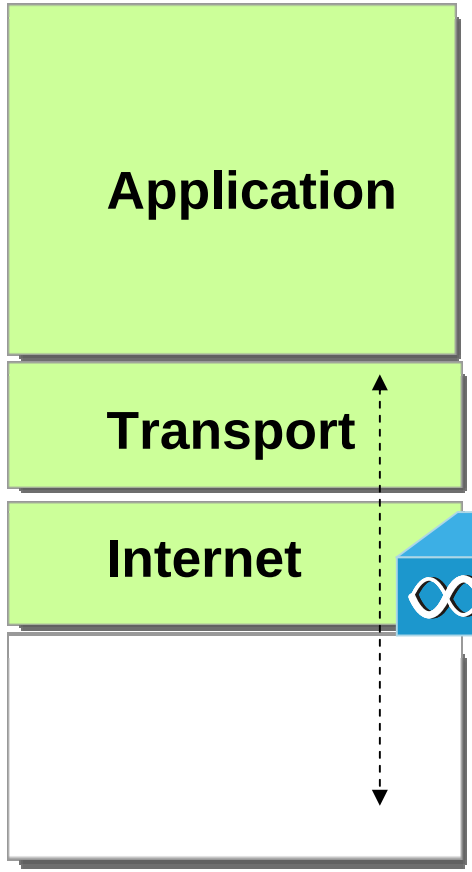
Firewalls

Screening Firewalls and Proxies:

Proxy - isolates local network from untrusted networks (AKA: Application gateway)

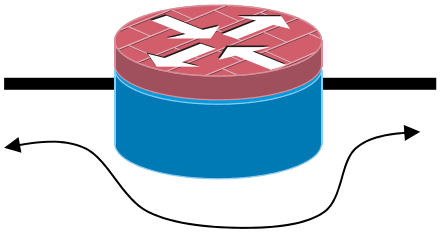






Internet model

Screening firewall: Filters for source and destination TCP ports



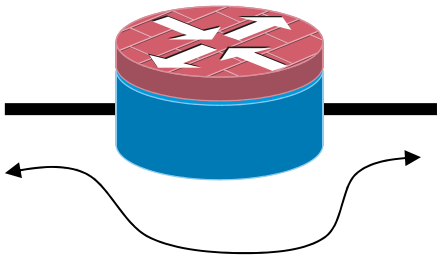
Screen firewall: Filters for source and destination IP addresses

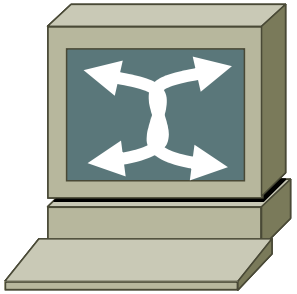
Author: Bill Buchanan

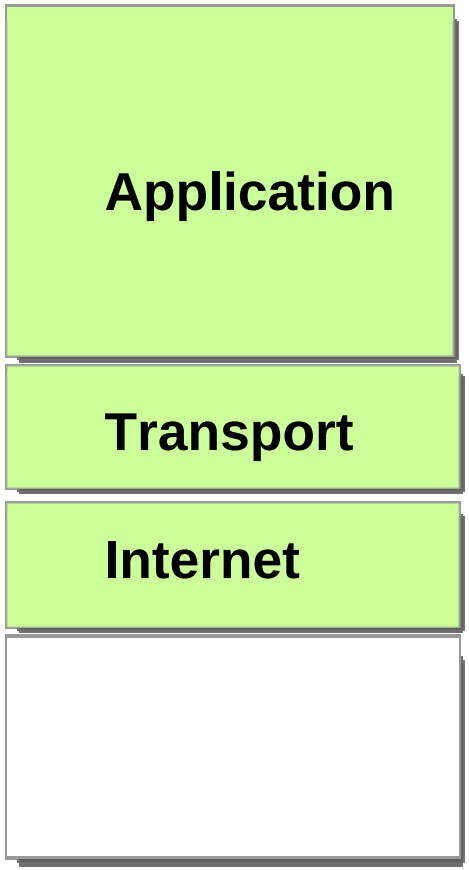
Firewalls and Proxies

Screening Firewalls and Proxies:

Proxy - isolates local network from untrusted networks (AKA: Application gateway)







Internet model

Screening firewall:

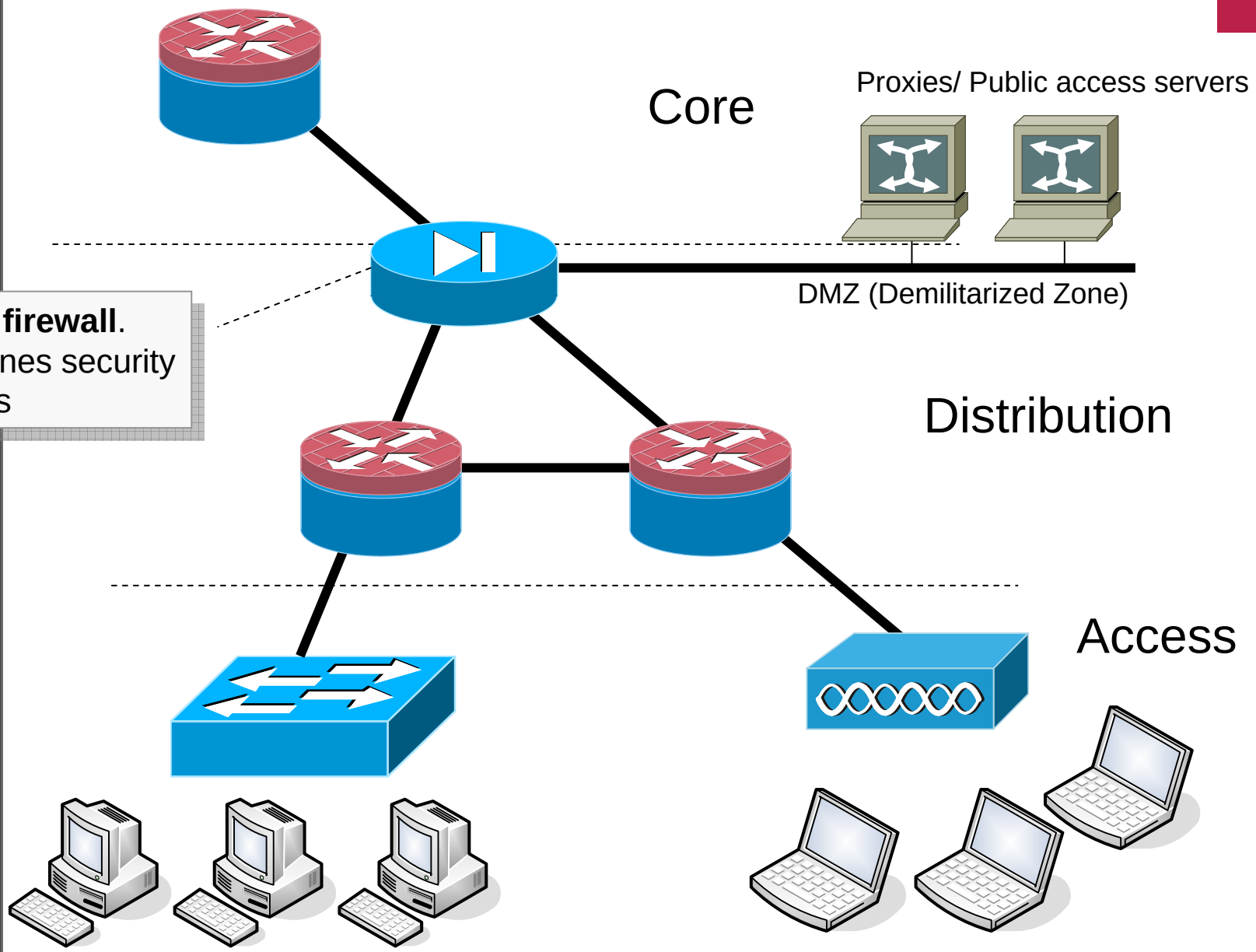
Advantages:

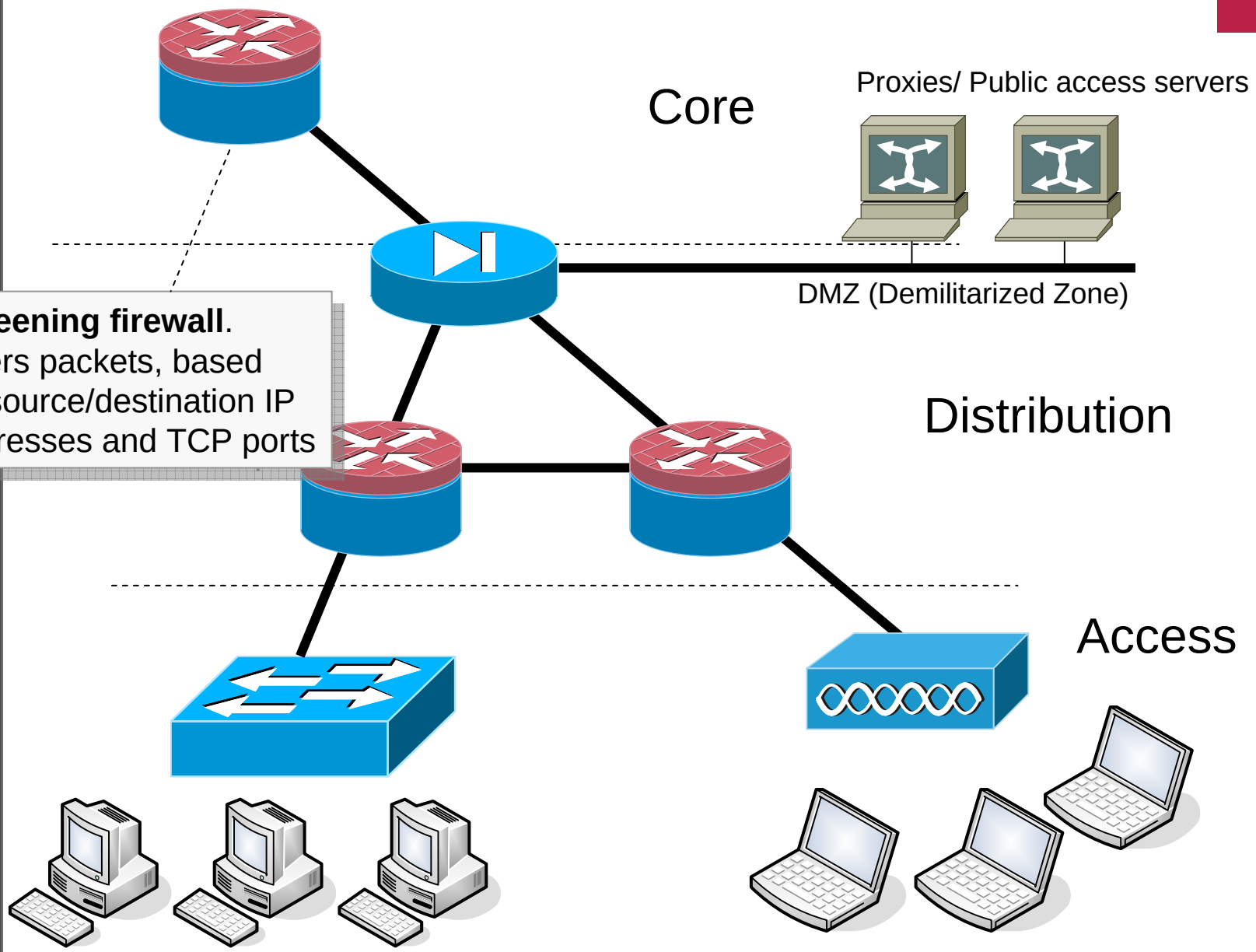
- Simple.
- Low costs

Disadvantages:

- Complexity of rules.
- Cost of managing firewall.
- Lack of user-authentication.

Core
Proxies/ Public access servers
DMZ (Demilitarized Zone)
Distribution
Access

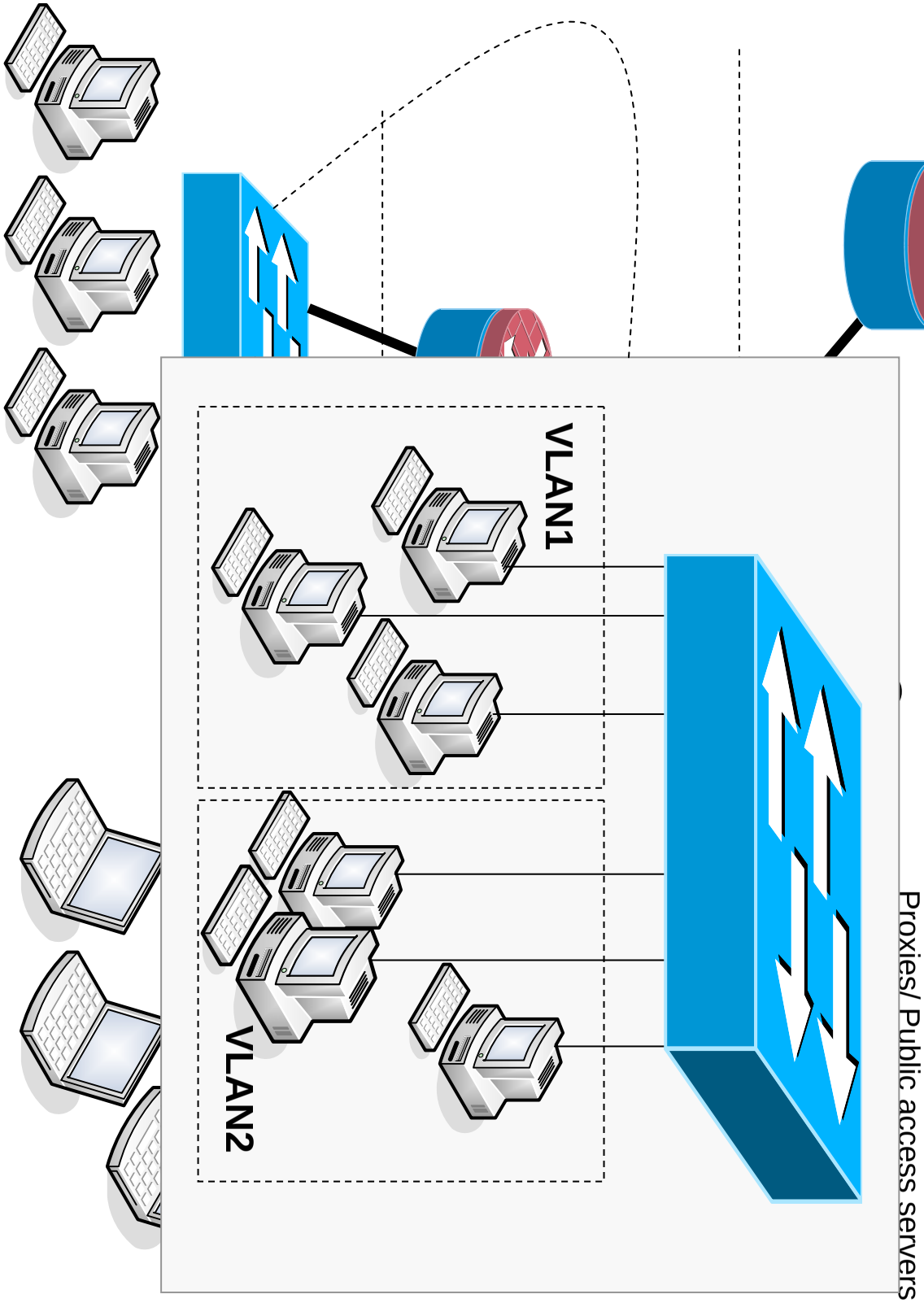

Core
Proxies/ Public access servers
DMZ (Demilitarized Zone)
PIX firewall.
Defines security rules
Distribution
Access



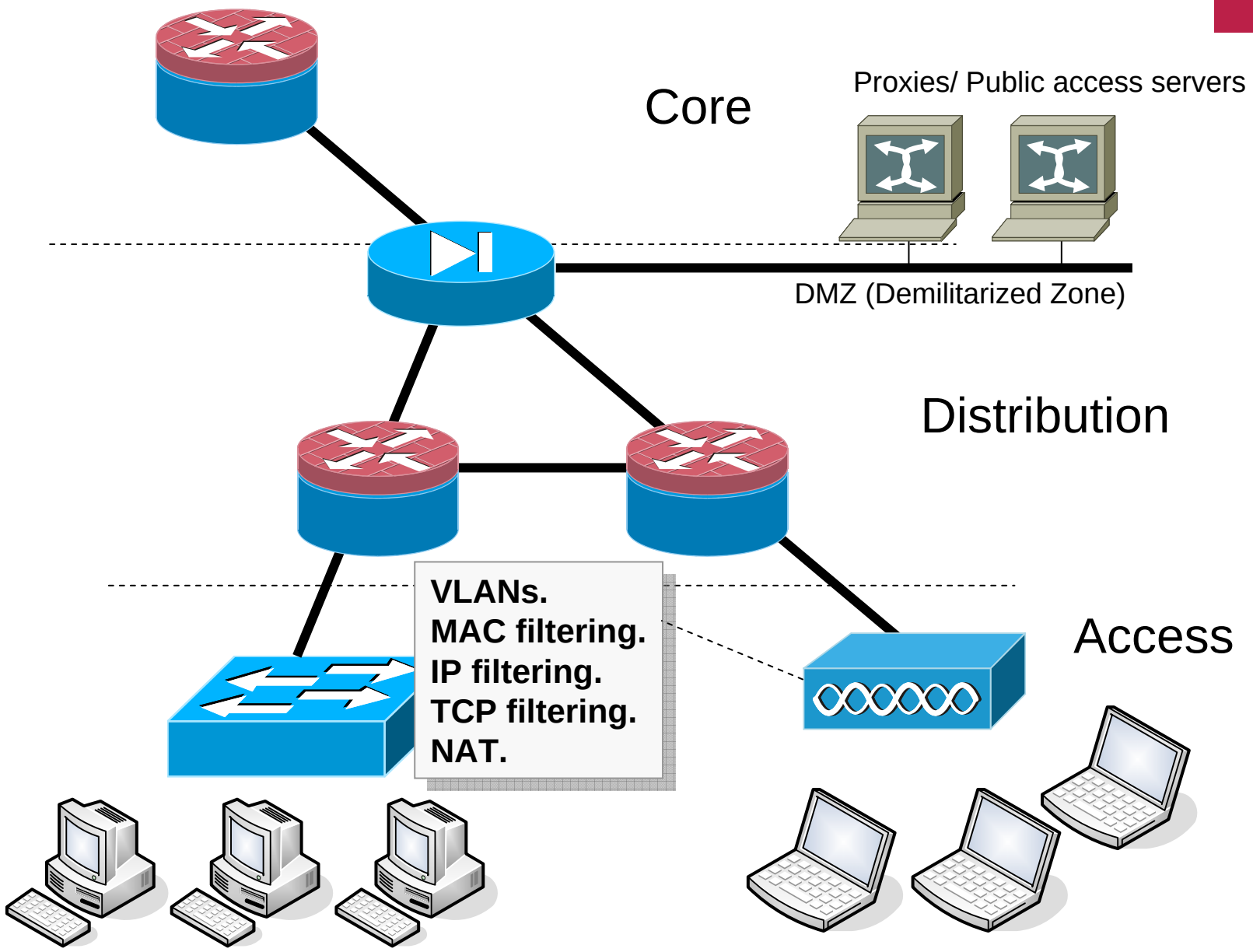
Screening firewall.
Filters packets, based on source/destination IP addresses and TCP ports

Wireless LAN - CO72047

Prof W Buchanan - Centre
for Dist. Computing and Security



Author: Bill Buchanan

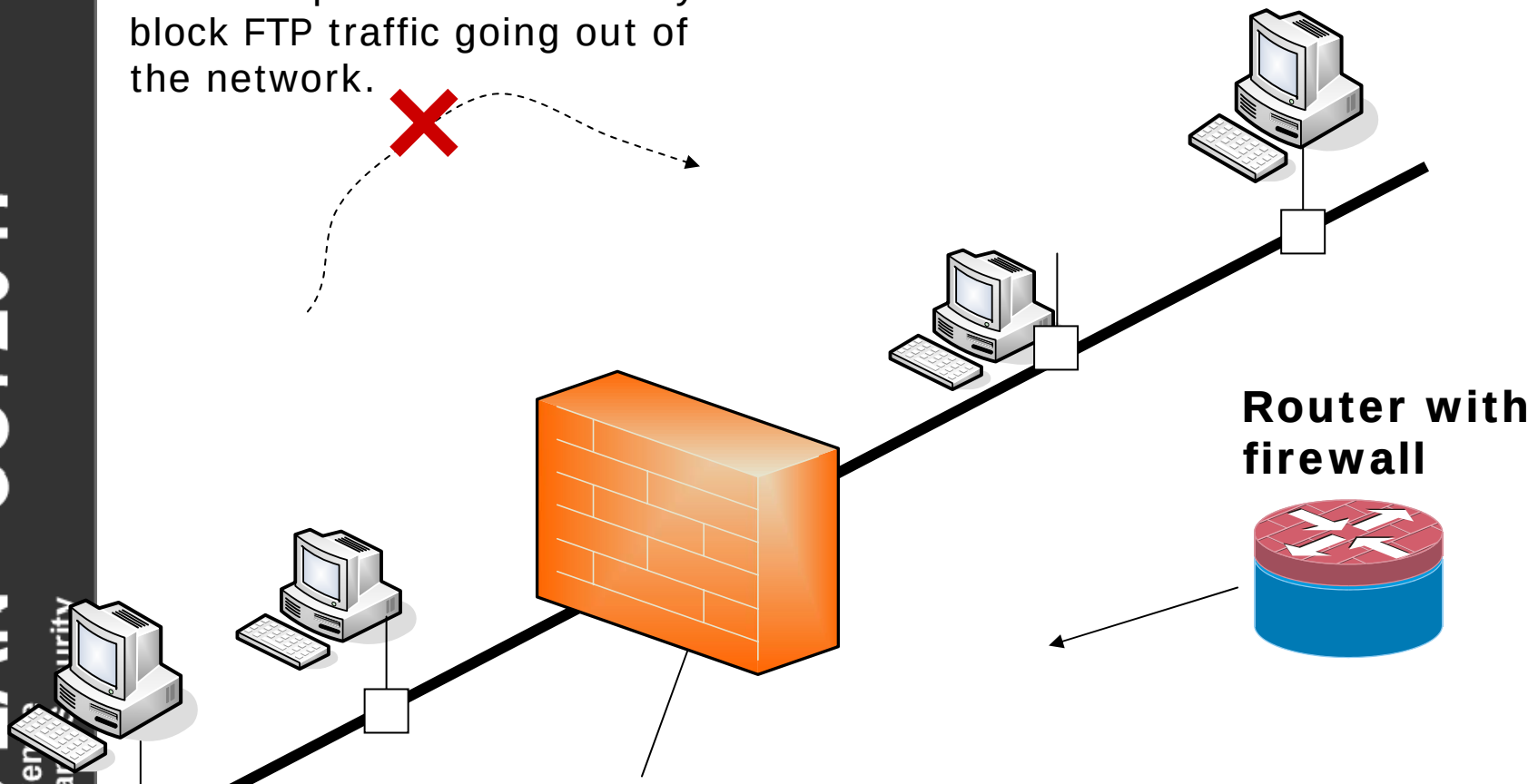

Core
Proxies/ Public access servers
DMZ (Demilitarized Zone)
Distribution
Access
VLANs.
MAC filtering.
IP filtering.
TCP filtering.
NAT.

Screening Firewall

Author: Bill Buchanan

Screening Firewall

For example the firewall may block FTP traffic going out of the network.



A **port** on a router can be setup with **ACLs** to filter traffic based on the network address or the source or destination port number

ACLs

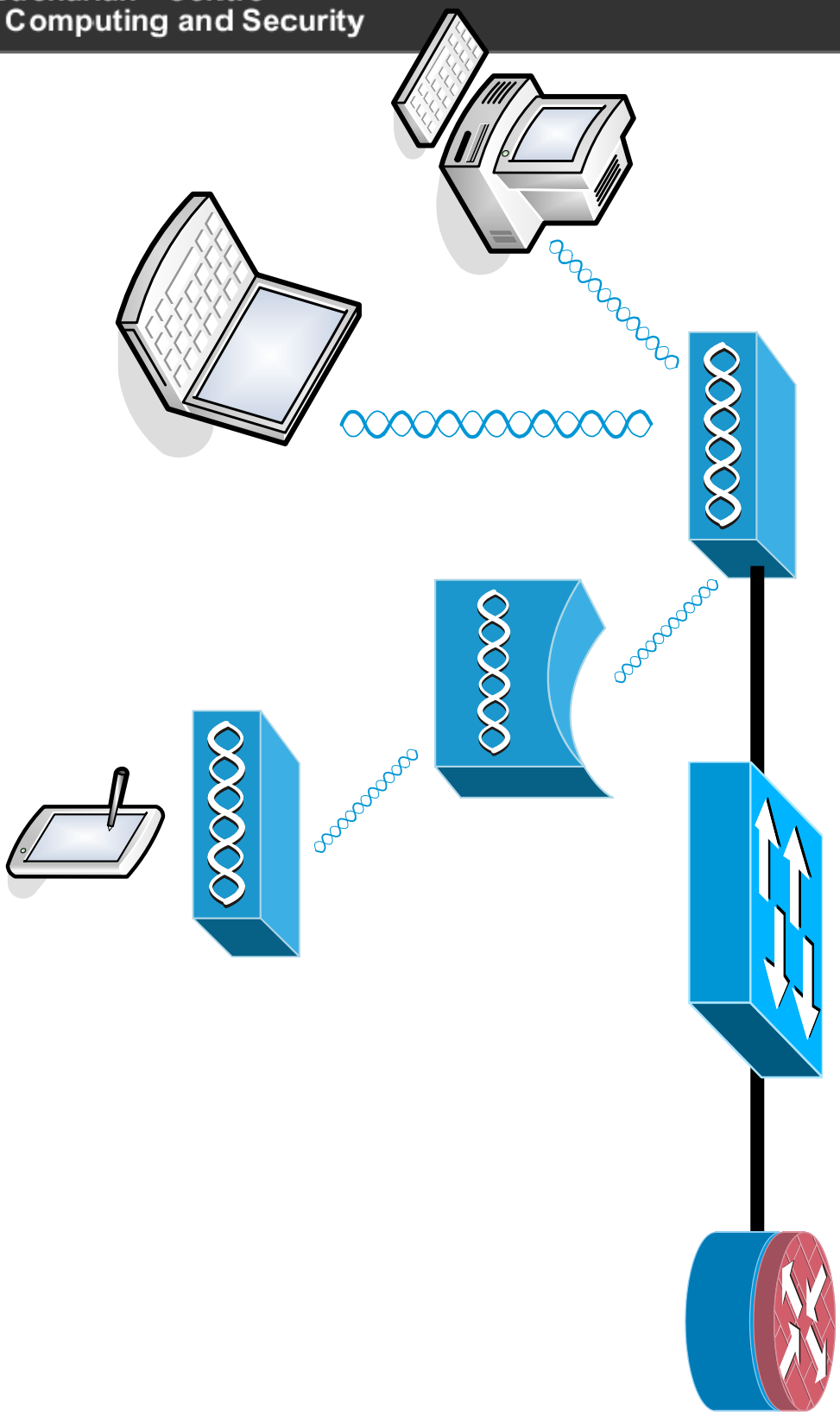
- **MAC address.**
- **Source IP address**. The address that the data packet was sent from.
- **Destination IP address**. The address that the data packet is destined for.
- **Source TCP port**. The port that the data segment originated from. Typical ports which could be blocked are FTP (port 21), TELNET (port 23), and WWW (port 80).
- **Destination TCP port**. The port that the data segment is destined for.
- **Protocol type**. This filters for UDP or TCP traffic.

MAC address filtering

Author: Bill Buchanan

Wireless LAN - CO72047

Prof W Buchanan - Centre
for Dist. Computing and Security



Author: Bill Buchanan

Scope of MAC address filtering

Defined by broadcast domain

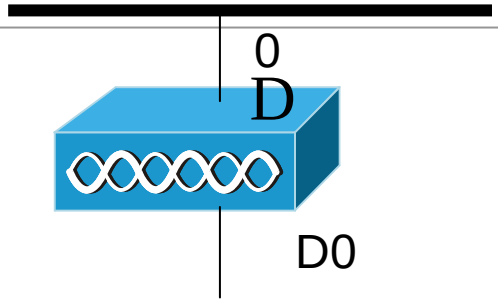
Author: Bill Buchanan

```
access-list [<700-799> | <1100-1199>] [deny | permit] [source
ac] [source mask] [dest mac] [dest mask]
```

For example to disallow the node with the mac address of 0090.4b54.d83a access to 0060.b39f.cae1:

```
(config)# access-list 1101 deny  0090.4b54.d83a 0.0.0
0060.b39f.cae1 0.0.0
(config)# access-list 1101 permit 0.0.0  ffff.ffff.ffff 0.0.0
ffff.ffff.ffff

(config)# int d0
(config-if)# l2-filter bridge-group-acl
(config-if)# bridge-group input-address-list 1101
```



Standard ACLs

Author: Bill Buchanan

Standard ACLs

```
Router# access-list access-list-value {permit | deny} source source-mask
```

```
Router# access-list 1 deny 156.1.1.10 0.0.0.0
```

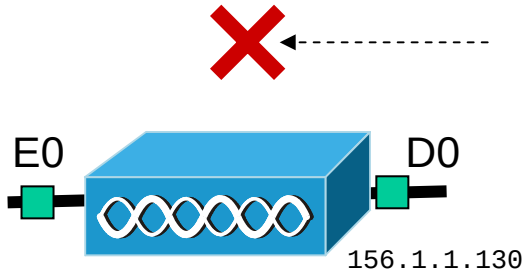
```
Router# access-list 1 deny 156.1.1.0 0.0.0.255
```

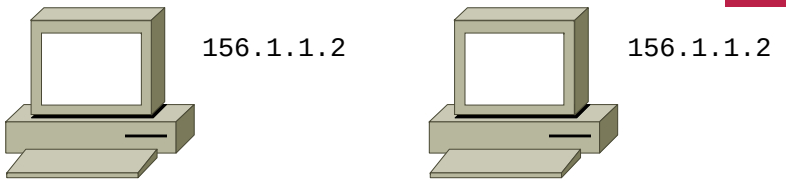
```
Router# access-list 1 deny 156.1.1.0 0.0.0.255
Router# access-list 1 permit ip any any
```

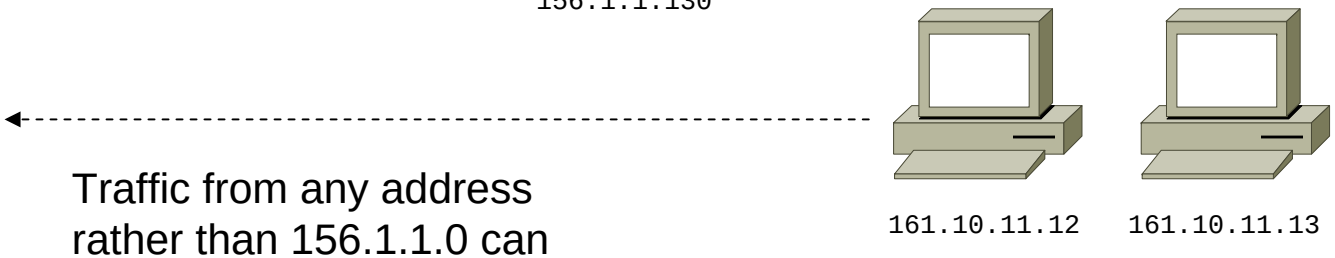
Standard ACLs filter on the source IP address

```
Router (config)# interface Ethernet0
Router (config-if)# ip address 156.1.1.130 255.255.255.0
Router (config-if)# ip access-group 1 in
```

Standard ACLs







Traffic from any address rather than 156.1.1.0 can pass

Match this part

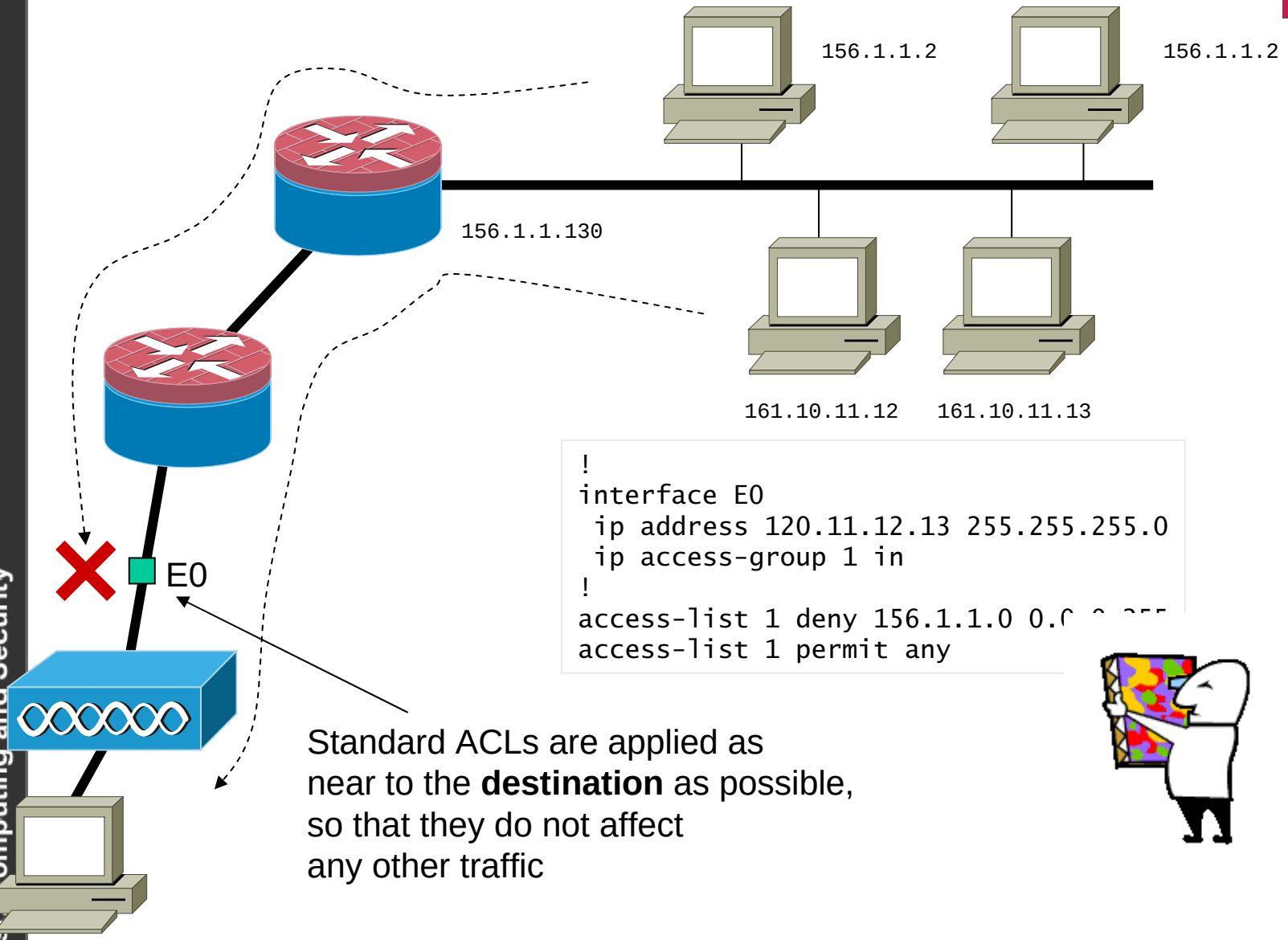
```
Router# access-list 1 deny 156.1.1.0 0.0.0.255
Router# access-list 1 permit any
```

Ignore this part

```
Router (config)# interface D0
Router (config-if)# ip address 156.1.1.130 255.255.255.0
Router (config-if)# ip access-group 1 in
```

Author: Bill Buchanan

Standard ACLs




```
!
interface E0
 ip address 120.11.12.13 255.255.255.0
 ip access-group 1 in
!
access-list 1 deny 156.1.1.0 0.0.0.255
access-list 1 permit any
```


Standard ACLs are applied as near to the **destination** as possible, so that they do not affect any other traffic

Author: Bill Buchanan

```
 (config)#ip access-list standard ?
  <1-99>        Standard IP access-list number
  <1300-1999>   Standard IP access-list number (expanded range)
  WORD          Access-list name
where WORD is the name of the access-list is be defined. For example:

(config)#ip access-list standard Test

(config-std-nacl)#?
Standard Access List configuration commands:
  deny     Specify packets to reject
  exit     Exit from access-list configuration mode
  no       Negate a command or set its defaults
  permit   Specify packets to forward
and to define a standard access-list:
 (config-std-nacl)#deny 156.1.1.0 0.0.0.255
 (config-std-nacl)#permit ?
  Hostname or A.B.C.D  Address to match
  any                  Any source host
  host                 A single host address
```

```
(config-std-nacl)#permit ?
  Hostname or A.B.C.D  Address to match
  any                  Any source host
  host                 A single host address
 (config-std-nacl)#permit any ?
  log  Log matches against this entry
  <cr>
(config-std-nacl)#permit any

It can then be applied with:
(config)#int e0
(config-if)#ip access-group ?
  <1-199>      IP access list (standard or extended)
  <1300-2699>  IP expanded access list (standard or extended)
  WORD         Access-list name
(config-if)#ip access-group Test ?
  in   inbound packets
  out  outbound packets
(config-if)#ip access-group Test in
```

Extended ACLs

Author: Bill Buchanan

Extended ACLs

```
Router# access-list access-list-value {permit | deny} {test-conditions}
```

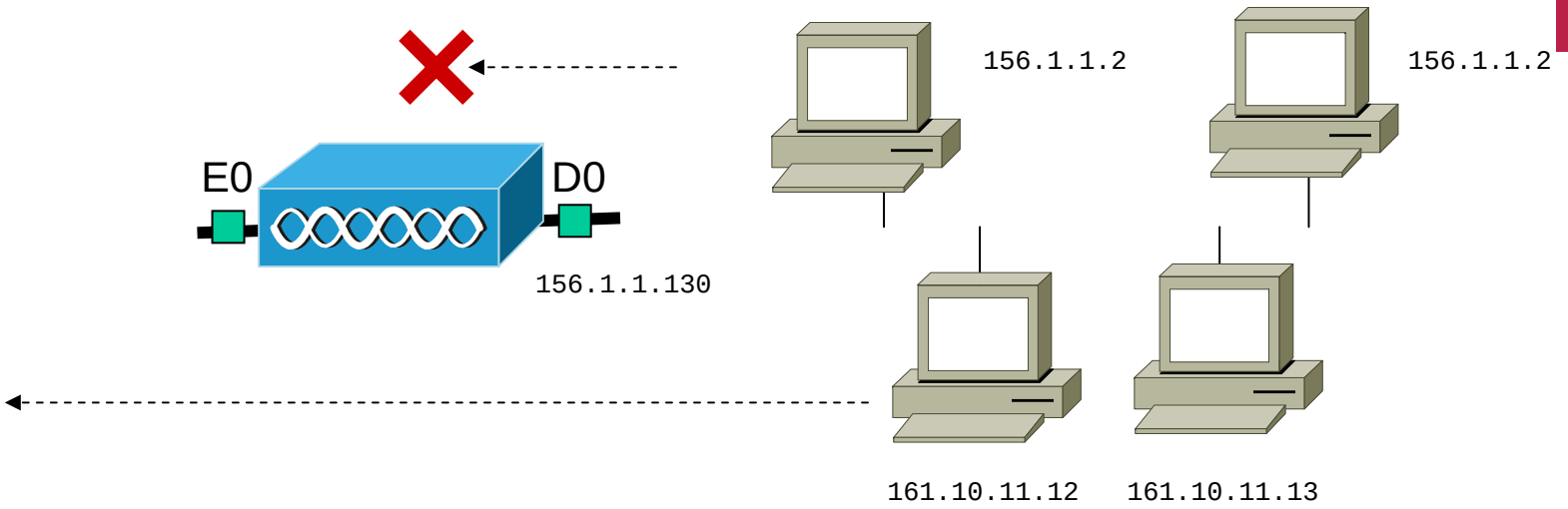
```
Router(config)#access-list 100 deny ip host 156.1.1.134 156.70.1.1 0.0.0.0
Router(config)#access-list 100 permit ip any any
```

```
Router(config)#access-list 100 deny ip 156.1.1.0 0.0.0.255 156.70.1.0
                                                    0.0.0.255
Router(config)#access-list 100 permit ip any any
```

```
Router(config)#access-list 100 deny ip 156.1.1.0 0.0.0.254 host 156.70.1.1
Router(config)#access-list 100 permit ip any any
```

```
Router (config)# interface Ethernet0
Router (config-if)# ip address 156.1.1.130 255.255.255.192
Router (config-if)# ip access-group 100 in
```

Extended ACLs



from to

```
(config)#access-list 100 deny ip host 156.1.1.2 70.1.2.0 0.0.0.255
(config)#access-list 100 permit ip any any
```

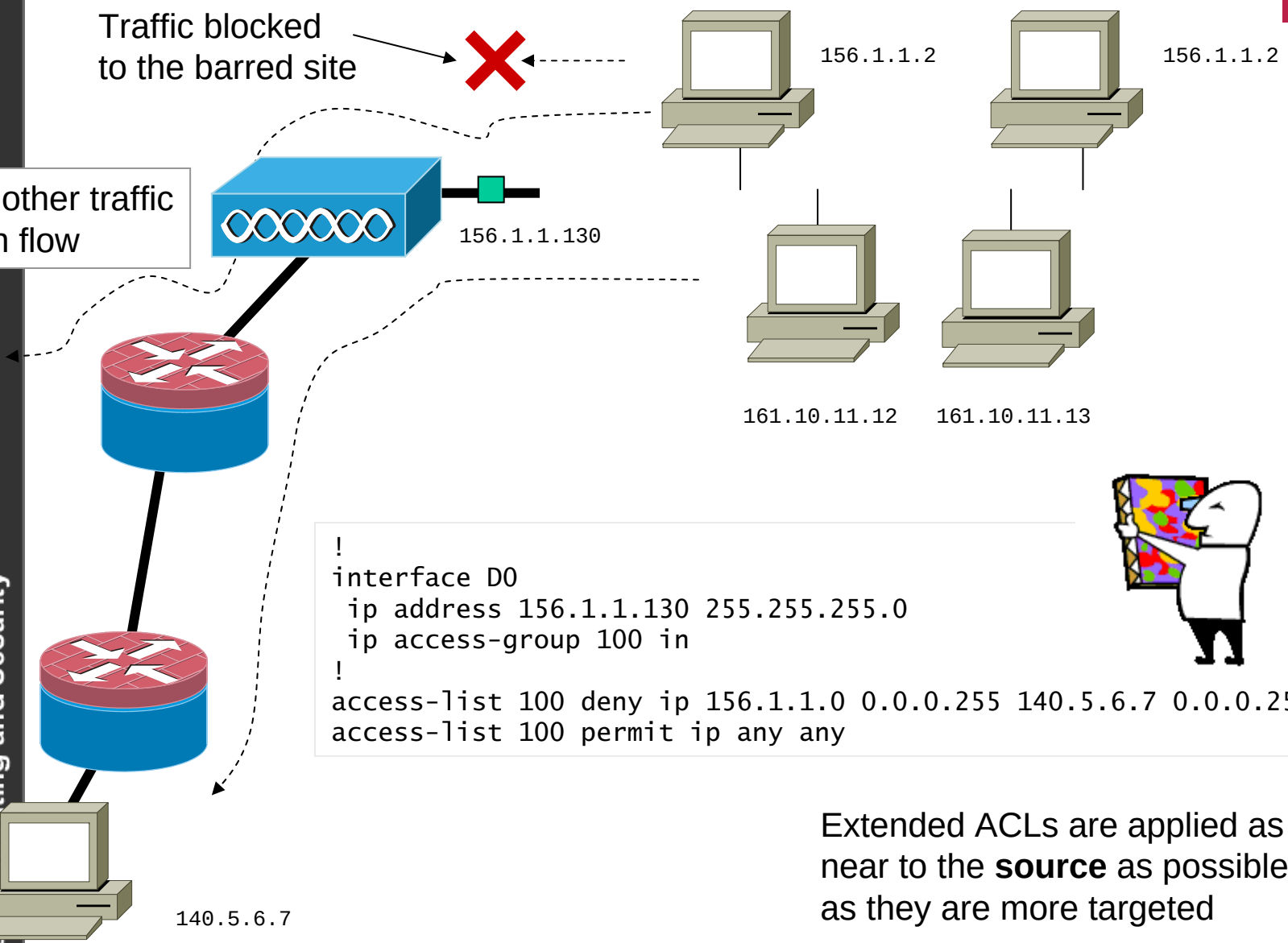
Denies traffic from 156.1.1.2 to the 70.1.2.0 network

```
(config)#access-list 100 deny ip 156.1.1.0 0.0.0.255 70.1.2.0 0.0.0.255
(config)#access-list 100 permit ip any any
```

Denies traffic from any host on 156.1.1.0 to the 70.1.2.0 network

Author: Bill Buchanan

Example of an Extended ACL




```
!
interface D0
 ip address 156.1.1.130 255.255.255.0
 ip access-group 100 in
!
access-list 100 deny ip 156.1.1.0 0.0.0.255 140.5.6.7 0.0.0.255
access-list 100 permit ip any any
```


Extended ACLs are applied as near to the **source** as possible, as they are more targeted

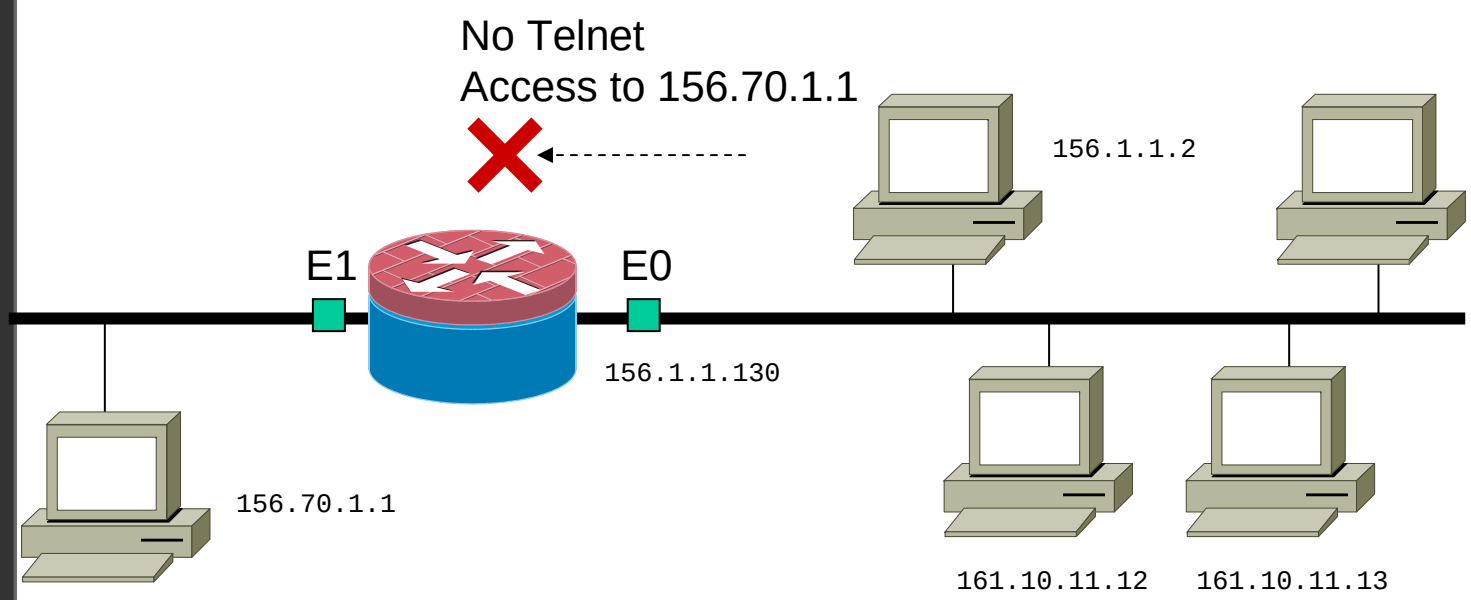
Extended ACLs filtering TCP traffic

An extended ACLs can also filter for TCP/UDP traffic, such as:

Optional field in brackets

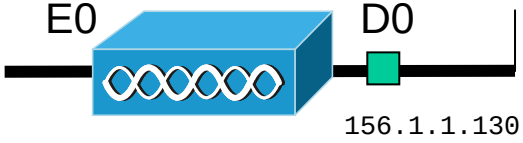
```
Router(config)#access-list access-list-value { permit | deny } {tcp | udp
      | igrp} source source-mask destination destination-mask {eq | neq | lt |
      gt} port
```

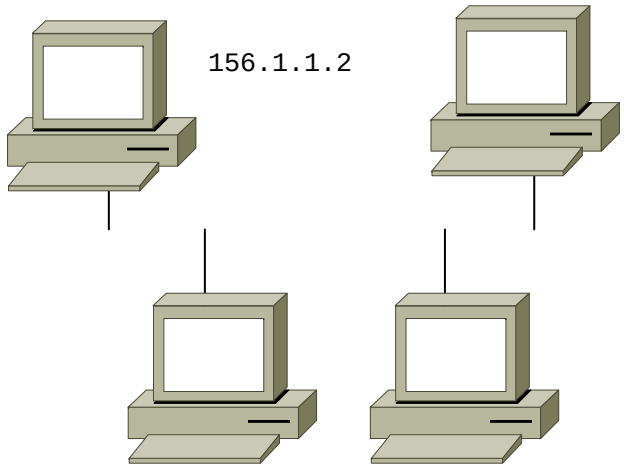
```
access-list 101 deny tcp 156.1.1.0 0.0.0.255 eq any host 156.70.1.1 eq telnet
access-list 101 permit ip any any
```



Open and closed firewalls

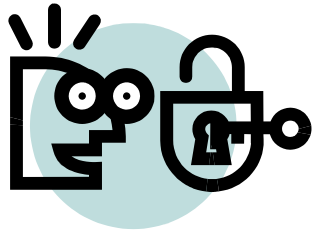
```
access-list 101 permit ….
access-list 101 deny ip any any
```

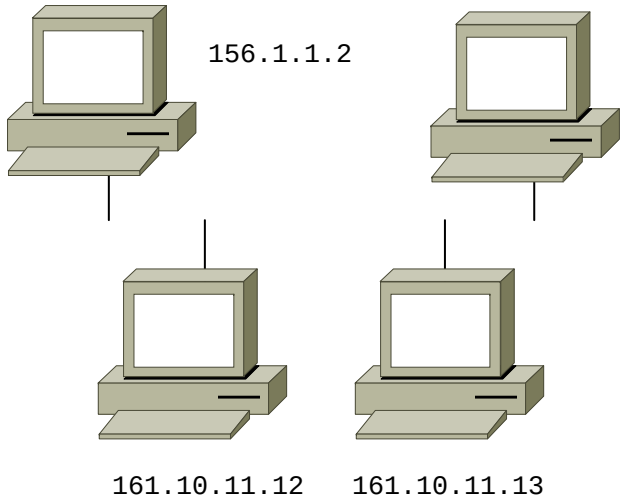




A **closed** firewall, permits some things, and denies everything else

```
access-list 101 deny ….
access-list 101 permit ip any any
```





An **open** firewall, denies some things, and permits everything else

Author: Bill Buchanan