



IS
2nd
material

Network



عن هذا التلخيص

1. في الملخص أحاول عدم الخروج عن محتوى المحاضرة إلا في الضرورة وفي هذه الحالة:
 1. الملاحظات المائلة هكذا تكون شرحا ليس مكتوبا في المحاضرة نفسها
 2. إذا هناك صفحة بكاملها أو جزء بكامله ليس في المحاضرة نفسها بل من فهمي سأوضح ذلك
 2. الأمثلة أختصرها هنا فلا آتي بها كاملة إلا في الضرورة
 3. غالبا لن أستخدم نفس عبارات المحاضرة فإذا كنت تريد نفس عبارات الحاضرة لكي تحفظها بالضبط كما هي تحضيراً لامتحان معين أو ما شابه يمكنك قراءة التلخيص ثم المرور على ملف المحاضرة سريعا
 4. المحتوى أنا مضطر لجعله بالإنجليزية حتى أضمن عدم نسيان المصطلحات التي يمكن أن تأتي فيها أسئلة ... هذا مع الأسف: (... ولغتنا الجميلة أولى بأن نعطي لها اهتماما
-

chapter 1: introduction

roadmap (content) of this lecture

1. What is the Internet? What is a protocol?
2. Network edge: hosts, access network, physical media
3. Network core:
 - a. packet/circuit switching
 - b. internet structure
4. Performance:
 - a. loss,
 - b. delay,
 - c. throughput
5. Protocol layers, service models
6. Security
7. History

Goal :

1. understanding the "feel" and "big picture"
2. Introduction to terminology

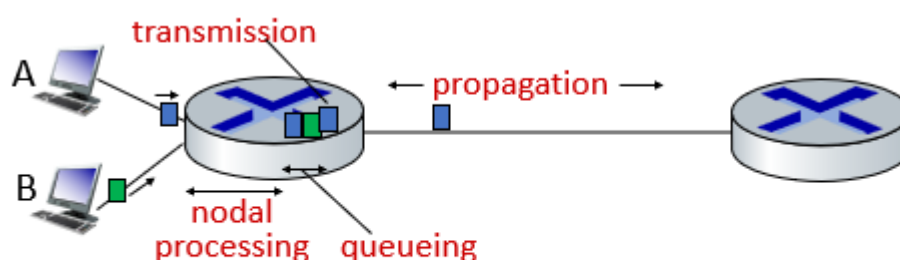
- *The content of this file is from the 4th point till the end of the lecture*

Performance: loss, delay, throughput

- Packet QUEUE in router buffers (internal memory) happens when the arrival rate to link (data going to the router) is higher than output link capacity (data going from the router) temporarily
- Packet LOSS occurs when the internal memory of the router is full (no more room for another packet)

Packet delay: four sources

The time (delay) a packet takes to be transmitted through one node (router) has 4 reasons:



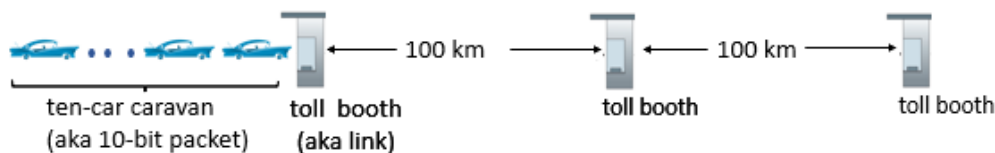
$$d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}$$

1. The over all delay = $d(\text{nodal})$
2. $d(\text{proc})$ is called "nodal processing" and it's less than microseconds... this delay is because we
 - a. check bit errors (see if packet data is right or wrong)

- b. determine output link (which router will this packet be sent to in the next step)
3. $d(\text{queue})$ is called queueing delay, depends on the congestion (تكديس) level of router, and it's equal to: "time waiting at output link for transmission"
4. $d(\text{trans}) = L/R = \text{transmission delay}$
 - a. L: packet length (in bits)
 - b. R: link transmission rate (in bps)
5. $d(\text{prop}) = \text{propagation delay} = d/s$
 - a. d: length of physical link
 - b. s: propagation speed ($\sim 2 \times 10^8$ m/sec)

notice that $d(\text{prop})$ is not equal to $d(\text{trans})$ But how ?? ... in the lecture there is an example that shows the difference .. I'll summarize it here in Arabic for easier understanding

Caravan analogy



- تخيل معي أن هناك قافلة من العربات تمشي على الطريق مع بعضها البعض وعددهم عشرة ... الطريق الذي تمشي عليه العربات فيه أكشاك لأخذ ضرائب من المسافرين ... بين كل كشك والذي بعده 100 كم.
- الآن تخيل أيضا أن المدة التي تستغرقها عربة واحدة لكي يخدمها الكشك = 12 ثانية والعربات تنتظر بعضها البعض حتى ينتهوا جميعا ثم ينطلقوا في الرحلة مع بعضهم حتى الكشك التالي وهكذا
- أخيرا تخيل أن كل عربة من هذه العربات سرعتها ثابتة بين الأكشاك = 100 كم لا تتغير

- بناء على تصورك للمشهد الآن تستنتج أن المدة التي يستغرقها كشك واحد حتى ينتهي من خدمة القافلة جميعها = 10 عربات * 12 ثانية = 120 ثانية ... والمدة التي تستغرقها القافلة حتى تقطع الطريق بين الأكشاك = 100 / 1 = 100 ساعة = 60 دقيقة (الزمن = المسافة / السرعة) ... إذا الزمن الكلي = 60 دقيقة + 2 دقيقة = 62 دقيقة

- كيف تربط بين هذا المثال وبين ما يحدث في الشبكات؟؟ كالتالي:

- العربة الواحدة: bit
- القافلة (caravan): packet
- الكشك (toll booth): router
- الطريق بين كل كشك والذي بعده: link
- الخدمة التي يقدمها الكشك (يأخذ ضرائب السفر يعني): link transmission service
- الزمن الذي يأخذه الكشك لإنهاء الخدمة لكل العربات (دقيقتان في المثال): transmission delay
- الزمن الذي يأخذه السفر على الطريق بين كل كشك والذي بعده (ساعة في المثال): propagation delay
- مسافة الطريق بين كل كشكين (100 كم في المثال) = d (length of physical link)
- سرعة السيارات على الطريق (100 كم كل ساعة في المثال) = s (propagation speed)
- وأخيرا لسنا محتاجين لحساب R أو L (packet length) (transmission rate) من المثال لأنه بالفعل ذكر الزمن مباشرة $d(trans) = L/R$ وهو الذي نريده

الآن تخيل أن الكشك يستغرق في السيارة الواحدة دقيقة كاملة وسرعة السيارات على الطريق بين كل كشكين = 1000 كم / ساعة ... إذا لم تكن العربات تمشي مع بعضها البعض ويمكن أن تسبق أحدها الأخرى: هل يمكن أن تصل بعد العربات للكشك رقم 2 في حين أن بعض العربات الأخرى لا تزال عند الكشك الأول؟

- نعم هذا صحيح! لكن كيف؟
 - العربة الأولى التي خدمها الكشك تأخذ دقيقة .. ثم تأخذ الطريق من الكشك الأول إلى الثاني في (1000 / 100) ساعة = 6 دقائق ... أي أنها تأخذ 1 + 6 = 7 دقائق
 - في هذه الأثناء بعد مرور 7 دقائق نرى أن 7 عربات قد تحركت بالفعل من الكشك الأول وباقي 3 لا تزال لم تتحرك
 - إذا نستخلص من هذا كله أنه في 7 دقائق أول عربة وصلت للكشك الثاني بالفعل في حين أن هناك 3 عربات أخرى كانت لا تزال عند الكشك الأول

Packet queueing delay (revisited)

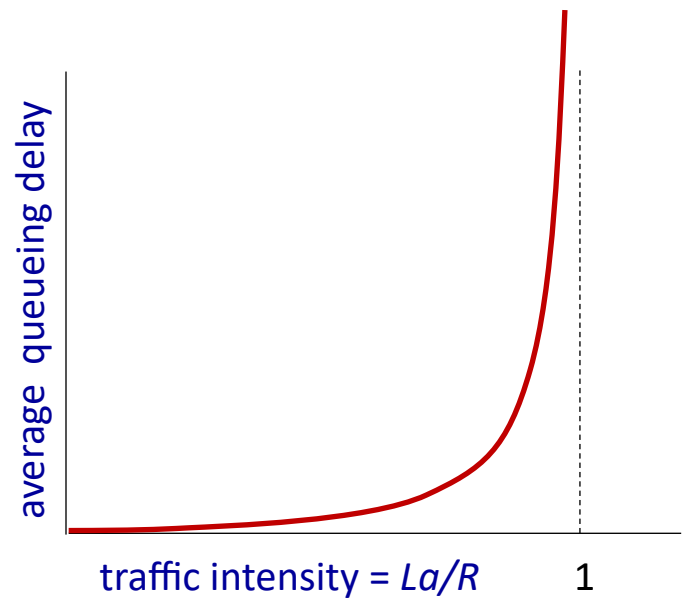
This is the “d(queue)” mentioned earlier ... it’s affected by something called “traffic intensity”

- traffic intensity can be calculated by this formula:

$$\frac{L \cdot a}{R} \quad ; \quad \frac{\text{arrival rate of bits}}{\text{service rate of bits}}$$

- a: average packet arrival rate
- L: packet length (bits)
- R: link bandwidth (bit transmission rate)
- The relation between traffic intensity and d(queue) is showed in the following figure:

- If traffic intensity (La/R) is close to ZERO: the average d(queue) aka (also known as) queueing delay is small
- If traffic intensity is close to ONE: d(queue) is large
- If traffic intensity is more than one: d(queue) is infinity .. that's because the amount of work that is required is more than what can be done actually



In the following lines, I will try to explain the concept of “traffic intensity” with an example in Arabic ... you can skip this part if you want ... this is NOT a part of the lecture .

This example is crafted with the help of ChatGPT.

- تخيل معي أن هناك مجموعة من الأشخاص يريدون تقديم هدية لشخص مهم يعيش في قصر
- تخيل أن a هو عدد الأشخاص الذين يدخلون القصر في كل دقيقة. كل شخص يمثل $packet$ (حزمة بيانات) تدخل الشبكة. الآن، تخيل أن L يمثل حجم الهدايا التي يحملها الزوار. كل زائر يحمل هدية تمثل $bits$ (بتات) من البيانات. بعض الزوار قد يحملون هدايا صغيرة (بتات قليلة)، بينما الآخرون يحملون هدايا أكبر (بتات كثيرة).
- (معدل الوصول) $= a \times L$ هو كمية البيانات التي تدخل القصر في كل دقيقة. يتم حسابها عن طريق عدد الزوار (a) مضروباً في حجم الهدايا (L) التي يحملونها. كلما زاد عدد الزوار أو كبرت الهدايا التي يحملونها، زادت كمية البيانات (عدد الـ $packets$) التي يجب أن يتعامل معها الراوتر.
- الزوار (a) هم الأشخاص الذين يدخلون القصر كل دقيقة. بعضهم يأتي بأعداد كبيرة (مثل دخول مجموعة كبيرة من الأشخاص)، مما يمثل عدد الـ $packets$ التي تصل إلى الراوتر. كل زائر يحمل هدية بحجم معين. إذا كانت الهدايا صغيرة، فالأمر بسيط (عدد قليل من الـ $bits$). لكن إذا كانت الهدايا كبيرة، فهذا يزيد من الضغط على الشبكة، لأن كل زائر يحتاج إلى وقت أكثر ليتمكن العاملون (الراوتر) من توجيهه داخل القصر.

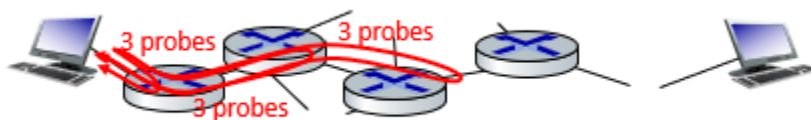
- العاملون في القصر (μ) هم الذين ينقلون الناس ويضعون الهدايا في أماكنها. كلما كانوا أسرع (μ أعلى)، كلما تمت معالجة الـ *packets* بشكل أسرع.
- إذا جاء عدد كبير من الزوار (a) ومعهم هدايا كبيرة (L)، يصبح القصر مزدحمًا (مثل الشبكة التي تستقبل كمية كبيرة من الـ *packets*). إذا كان العمال في القصر بطيئين (μ منخفضة)، سيبدأ الزوار بالتكدس في المداخل، مما يعني ازدحام الشبكة. شدة الازدحام $\rho = \lambda / \mu$: يعبر عن مدى ازدحام القصر؛ إذا كانت القيم كبيرة جدًا، القصر سيصبح مكتظًا، ويبدأ الزوار بالتأخر في الدخول (وهذا يشبه تأخر البيانات في الشبكة).
- a هو عدد الزوار (عدد الـ *packets*). L هو حجم الهدايا (عدد الـ *bits* في كل *packet*). λ هو حركة الزوار وحجم ما يحملونه معًا (معدل الوصول). μ هو سرعة العمال في ترتيب الهدايا (معدل معالجة الـ *packets*).
- إذا كانت الزوار يدخلون بسرعة ويحملون هدايا كبيرة، والقصر (الراوتر) بطيء في ترتيبهم، سيتكدسون، مما يعني أن الشبكة مزدحمة!

“Real” Internet delays and routes

How to measure delay and loss of packages in the REAL world?

There is a program called (traceroute) this program can do that by the following:

1. Assume that your device (sender) wants to send a message to another device (destination) and there are some routers between these two devices
2. Sending 3 packets to a router
3. The router returns them again to the sender (*your computer*)
4. *your device* (sender) measures the time each of the three packets took to be sent and received from this router
5. *Your computer* does that again and again with every router between it and the destination until



The output of the program will be like this:

```

1 cs-gw (128.119.240.254) 1 ms 1 ms 2 ms
2 border1-rt-fa5-1-0.gw.umass.edu (128.119.3.145) 1 ms 1 ms 2 ms
3 cht-vbns.gw.umass.edu (128.119.3.130) 6 ms 5 ms 5 ms
4 jn1-at1-0-0-19.wor.vbns.net (204.147.132.129) 16 ms 11 ms 13 ms
5 jn1-so7-0-0-0.wae.vbns.net (204.147.136.136) 21 ms 18 ms 18 ms
6 abilene-vbns.abilene.ucaid.edu (198.32.11.9) 22 ms 18 ms 22 ms
7 nycm-wash.abilene.ucaid.edu (198.32.8.46) 22 ms 22 ms 22 ms
8 62.40.103.253 (62.40.103.253) 104 ms 109 ms 106 ms
9 de2-1.de1.de.geant.net (62.40.96.129) 109 ms 102 ms 104 ms
10 de.fr1.fr.geant.net (62.40.96.50) 113 ms 121 ms 114 ms
11 renater-gw.fr1.fr.geant.net (62.40.103.54) 112 ms 114 ms 112 ms
12 nio-n2.cssi.renater.fr (193.51.206.13) 111 ms 114 ms 116 ms
13 nice.cssi.renater.fr (195.220.98.102) 123 ms 125 ms 124 ms
14 r3t2-nice.cssi.renater.fr (195.220.98.110) 126 ms 126 ms 124 ms
15 eurecom-valbonne.r3t2.ft.net (193.48.50.54) 135 ms 128 ms 133 ms
16 194.214.211.25 (194.214.211.25) 126 ms 128 ms 126 ms
17 * * *
18 * * *
19 fantasia.eurecom.fr (193.55.113.142) 132 ms 128 ms 136 ms

```

Notice that in each line there is

1. A router name
2. An address for the router
3. 3 time measures ... one for each packet sent to that router ... (this time measure means that the packet is sent then received in that time)

Also notice that there are three stars in one of the lines (*) ... this means there is no response from that router ... the three packets didn't return again so there is no time recorded*

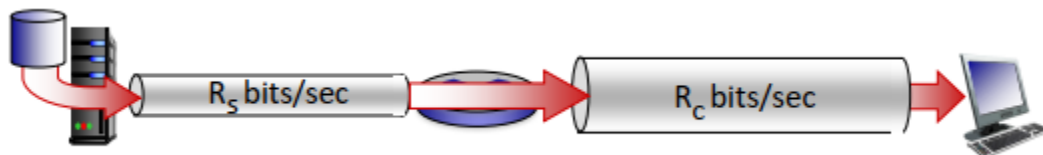
Throughput

It's the rate measured in (bits/time unit) at which bits are being sent from sender to receiver (سرعة نقل البيانات)

- We can measure throughput in two ways:
 - Instantaneous (لحظي): rate at a specific point in time
 - Average: rate over a period of time

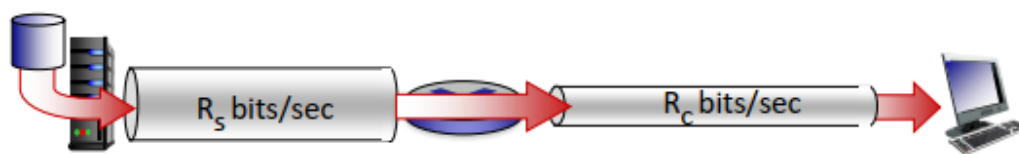
Imagine that you have a server, a client (normal device), and a router between them ... between the server and the router there is a link that has (throughput = R_s) ... between the client and the router there is a link that has (throughput = R_c).

- If $\rightarrow R_s < R_c$



○

- If $\rightarrow R_s > R_c$



○

in the second case ($R_s < R_c$) \rightarrow the link is called (bottleneck link) because it was big then small ... this makes the average throughput overall smaller than the first case ($R_s > R_c$)

bottleneck link

link on end-end path that constrains end-end throughput

- But how to calculate this average throughput we're talking about?
... it's equal to

Security

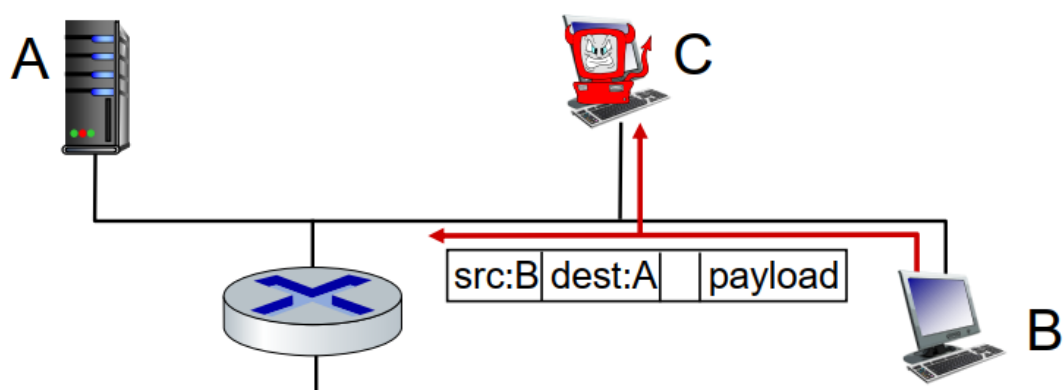
Internet was actually not designed to be secure ... but "internet protocol" designers worked to fix this ... there are three problems with security on the internet that we must think about:

1. How **BAD GUYS** can attack computer networks?
2. How we can **DEFEND NETWORKS** against attacks?
3. How to **DESIGN ARCHETICTURES** that are immune to attacks (مضادة للهجمات)

First, bad guys

They can do: packet interception (packet sniffing), fake identity (IP spoofing), denial of service (DoS)

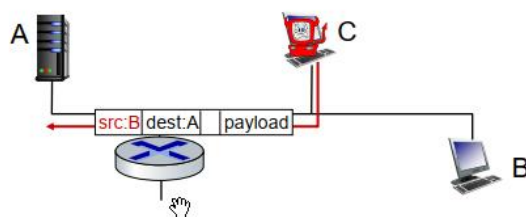
1. Packet interception = packet sniffing = (اعتراض الباكيت) .. = entering the network without permission and reading/recording all packets (including passwords and sensitive information) ... *the program that can do this for us is called (Wireshark)*



look at his face :)

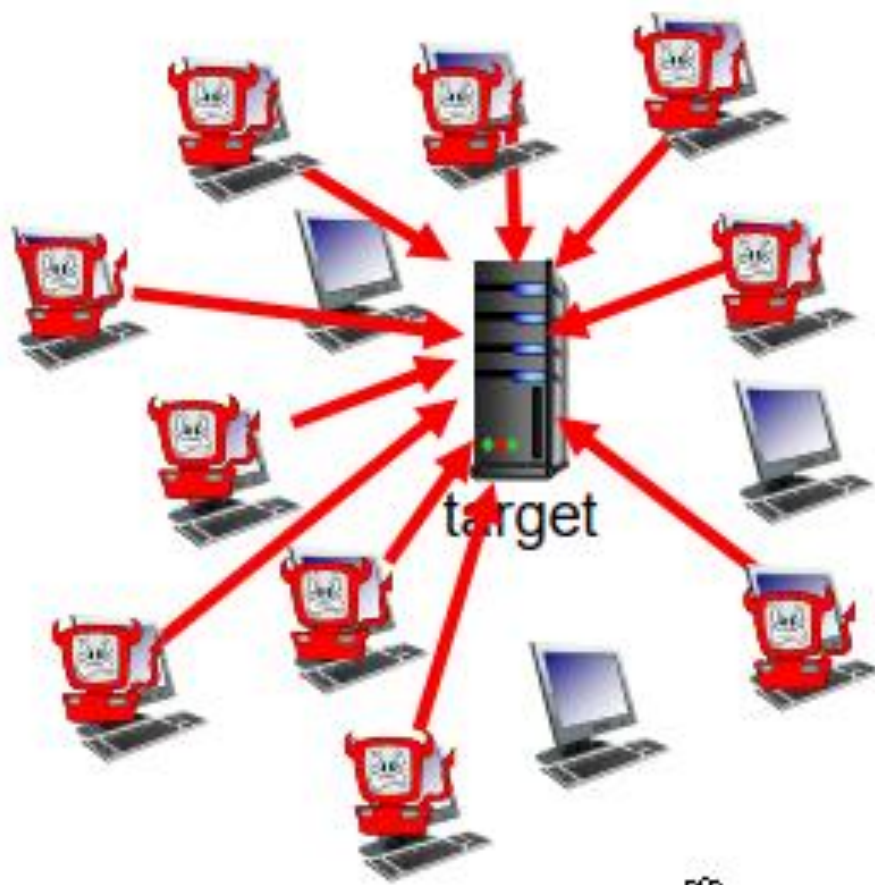
2. Fake identity = IP spoofing = (انتحال شخصية) = the bad guy takes an IP from the network and starts sending bad packets

IP spoofing: injection of packet with false source address



3. Denial of service = DoS = the bad guys (attackers) send a lot of requests to a server in the same time → this breaks down the server → good guys that was using the server will be harmed by this

Denial of Service (DoS): attackers make resources (server, bandwidth) unavailable to legitimate traffic by overwhelming resource with bogus traffic



Lines of defense

What can we do to stop the attackers from doing bad things??

Here are some examples:

- Authentication: identity check (*prove you are a good guy first before entering the system*)
- Confidentiality (الخصوصية أو السرية): this is done by “encryption”

- Integrity checks: preventing or just detecting tampering (التلاعب) of data
- Access restrictions: used in VPNs protected by passwords
- Firewalls: something in the middle of “network core” or “network access” (*we talked about these things in the previous summary*) ... here are some properties of “firewalls”
 - They are “off-by-default”: this means that they filter (*block*) incoming packets (من خارج الشبكة) to restrict senders, receivers, applications
 - It detect and react to a type of attacks called “DOS attacks”

Protocol layers, service models

Why do we organize the network into “layers”?

1. Networks contain many pieces ... this makes them complex ... and we want a simple way to organize it ... so we invented “protocol layers”
2. Overall, the system end-to-end (from to destination) involves many services ... each service is implemented by one layer ... and each layer depends on the previous one (below it)
3. Layering (splitting the network into layers) is an approach to design and discuss complex systems that makes it easy to:
 - a. Identify the different system pieces and relationships between them (فهم أجزاء النظام والعلاقة ما بينهم)

- b. Easy maintenance for the system (متابعة النظام سهلة) ... that's because a change in one layer doesn't affect the rest of the system

In the following lines we will discuss two systems to understand the concept of “layering” a network:


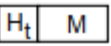
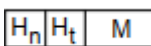
1. Layered internet protocol stack
2. ISO/OSI

Layered internet protocol stack

Before you read this: messages, segments, datagram, and frame are just different forms of data

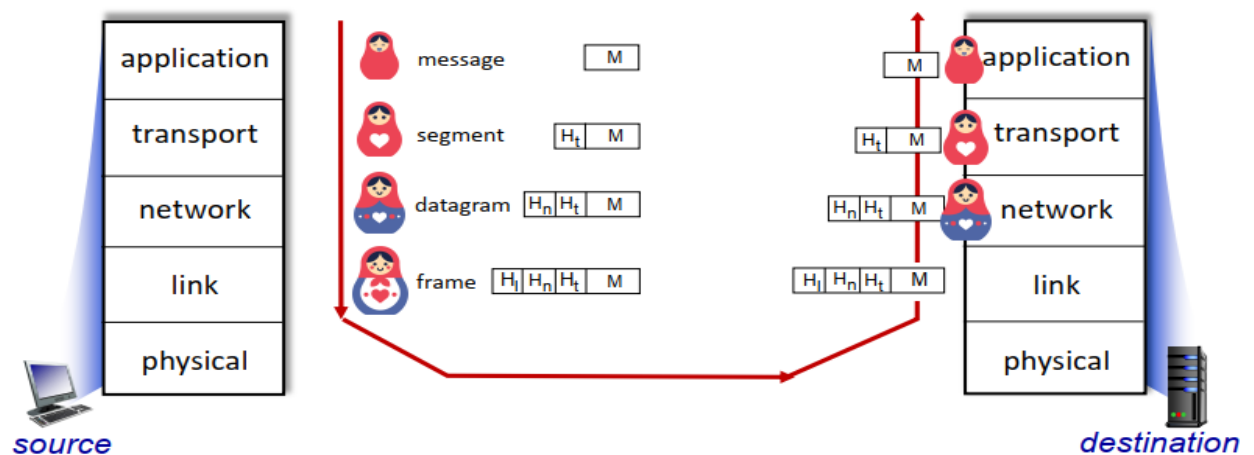
This protocol has 5 layers ...



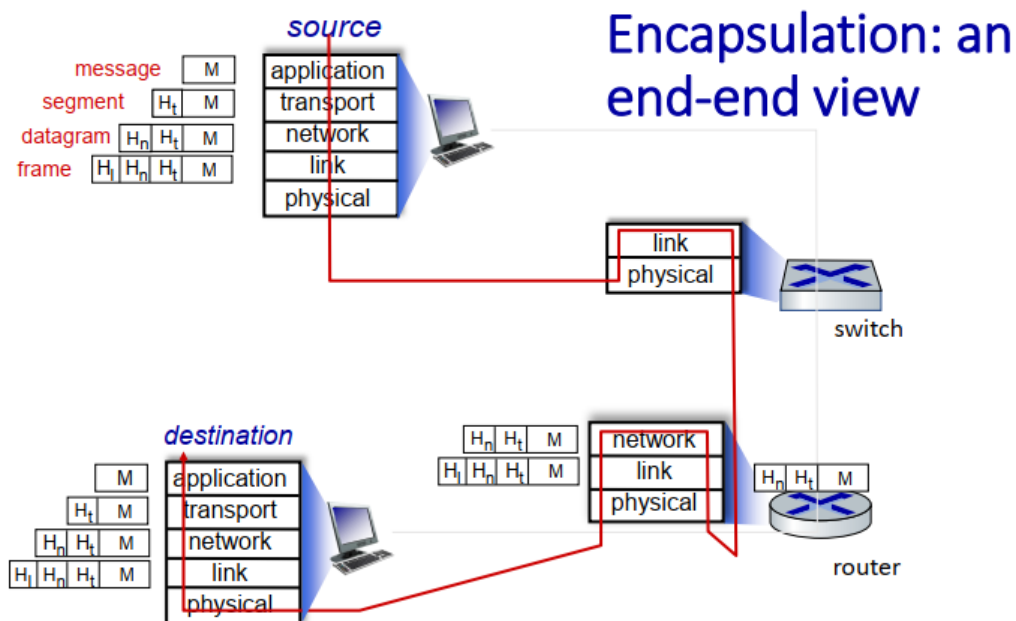
- APPLICATION:
 - Protocols used in this layer: HTTP, IMAP, SMTP, DNS
 - What does it do: exchange “messages” between “applications” 
- transport
 - Protocols: TCP, UDP
 - Do what? Transfer “segments” between “processes” by adding a “transport-layer header” to the “message” made in the application layer. 
- network:
 - Protocols: IP, routing protocols
 - Do what? Route (transfer) “datagrams” from the source all the way to the destination (الطريق كله من بداية المصدر حتى الهدف النهائي) and adds “network layer header” to the “segment” made in the transport layer 

- link:
 - Protocols: Ethernet, 802.11 (Wi-Fi), PPP
 - Do what? Transfer “frames” between neighboring elements (الأجهزة التي هي بجانب بعضها مباشرة) and adds “link layer header” to the “datagram” made in the network layer
- physical: bits “on the wire”

Services, Layering and Encapsulation



The application layer is implement using transport layer services which is implemented using network layer services which is implemented using link layer services ...



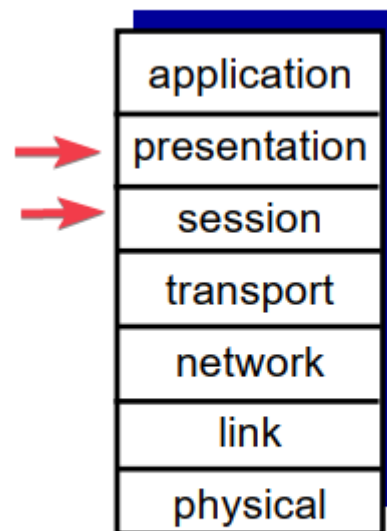
Notice that in this illustration:

The goal is to send the message (M) from the source to the destination ... this goal is done by encapsulating (تغليف) the data (M) then decapsulating it (إزالة الغلاف)

ISO/OSI reference model

Contains 2 additional layers (more than the internet protocol layer above)

1. presentation: allow applications to understand the meaning of data, (like for example: encryption, compression, machine-specific conventions)
2. session: synchronization, checkpointing, recovery of data exchange



The seven layer OSI/ISO reference model

But why doesn't the internet contain these two layers by default?? ... that's because if they are needed, they will be inside the "application" layer ... if they're not needed, nothing will change also (لا داعي لعمل طبقتين جديدتين طالما هناك طبقة بالفعل موجودة تنفذ هذه الخدمات)

History

This part is for reading only ... not coming in the exam